# Specifying Agents in Construct

Brian R. Hirshman, Kathleen M. Carley, Michael J. Kowalchuck

July 25, 2007
CMU-ISRI-07-107

Institute for Software Research
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

**Abstract**

Construct is a multi-agent simulation tool that is commonly used to investigate dynamic behavior in complex socio-cultural systems. This technical report describes the parameters necessary to specify agents in the simulation, focusing especially on the features which help describe agent behavior. It also introduces a number of pre-defined agent classes, stock-agents which can be used to quickly build up a simulation. This document is intended both as an introduction to Construct model and as a reference guide for simulation experts and casual modelers.

This work was supported in part by the Office of Naval Research (ONR N00014-06-1-0921, N00014-06-1-0104 and N00014-06-1-0772), the National Science Foundation (SES-0452487), the Army Research Lab (DAAD19-01-2-0011 ARL CTA 6.1 and DAAD19-01-2-0009 ARL CTA), and the AirForce Office of Sponsored Research (MURI: Cultural Modeling of the Adversary, 600322) for research in the area of dynamic network analysis. Additional support was provided by CASOS - the center for Computational Analysis of Social and Organizational Systems at Carnegie Mellon University. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Office of Naval Research, the National Science Foundation, the Army Research Lab or the U.S. government.

Version 1.1

# Report Documentation Page

*Form Approved*
*OMB No. 0704-0188*



| 1. REPORT DATE | 2. REPORT TYPE | 3. DATES COVERED |
|---|---|---|
| **25 JUL 2007** | | **00-00-2007 to 00-00-2007** |

| 4. TITLE AND SUBTITLE | 5a. CONTRACT NUMBER |
|---|---|
| **Specifying Agents in Construct** | 5b. GRANT NUMBER |
| | 5c. PROGRAM ELEMENT NUMBER |
| 6. AUTHOR(S) | 5d. PROJECT NUMBER |
| | 5e. TASK NUMBER |
| | 5f. WORK UNIT NUMBER |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Carnegie Mellon University,Institute for Software Research,School of Computer Science,Pittsburgh,PA,15213** | 8. PERFORMING ORGANIZATION REPORT NUMBER |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | 10. SPONSOR/MONITOR'S ACRONYM(S) |
| | 11. SPONSOR/MONITOR'S REPORT NUMBER(S) |

12. DISTRIBUTION/AVAILABILITY STATEMENT
**Approved for public release; distribution unlimited**

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT **unclassified** | b. ABSTRACT **unclassified** | c. THIS PAGE **unclassified** | **Same as Report (SAR)** | **69** | |

**Keywords:** construct, multi-agent simulation, dynamic network analysis, social network analysis, agents, agent modeling, agent class style sheet

# Contents

# List of Tables

# 1 Introduction & Motivation

In today's complex society, it is useful to predict and model the effect of a particular strategy before enacting it. Construct, a dynamic network evolution tool, can help address this issue by predicting how social behavior changes in response to the spread of information among individuals [1, 2]. Past work using Construct has examined the evolution of work groups, friendship networks, communication networks, and organizational social structure in response to information diffusion [1]. Future work using Construct will continue in these areas and expand into other areas as well.

To effectively use Construct, it is necessary to understand its simulation agents and their capabilities. Construct is an *agent-based* model, meaning that it consists of a series of computational entities in an interaction-knowledge space [1, 3]. These *agents* can represent the components of social network at various levels of fidelity; in some models, agents represents individuals in an interpersonal network, while in others agents represent teams, organizations, or corporations in much larger networks [1]. Many construct agents are designed to simulate humans or teams of humans, as described in later sections [4]. However, Construct agents need not be so: agents need only information storage and communication capabilities, so they can represent complex technological devices like web sites or databases [5]. One of Construct's greatest strengths is its ability to represent changing technology in many ways: while technology can amplify the abilities of existing agent, it can also be represented using entirely agents [4].

Though Construct agents have the power to interact in a single situation, Construct's true power comes from the fact that its agents can explore a variety of counter-factual environment changes. Construct can be used to run a series of *virtual experiments* using similar agents in order to examine how a diverse group of agents would behave in a slightly modified situation [3]. In a virtual experiment, model settings are largely fixed to represent a given state of the world while one of many *interventions*, or small modifications, are applied. By applying each intervention for multiple *runs*, or simulation iterations, the designer can search for successful strategies, explore possible side effects, and minimize experimental error [6]. At the conclusion of the virtual experiment, the designer can select the most effective strategy as the most favorable one to implement in a real-world environment [4, 6]. This virtual experiment technique has been used to evaluate phenomena such as group stability [7], consensus [8], informational diffusion [1], organizational adaptation [1, 4], and tax scheme propagation [5, 6].

This technical report provides additional detail on how to set up Construct agents, and by extension, Construct virtual experiments. In addition to discussing the meaning of various agent parameters, this report explains how these settings affect the behavior of an agent. The remainder of this document is organized as follows. Section 2 provides details about the types of parameters that can be set when specifying a Construct agent. Section 3 explains the notion of an agent class, an association of parameters whose specification can facilitate agent creation. Section 4 concludes.

# 2 Agent Parameters

The *parameters* of an agent are the values which define it behavior during the simulation. By modifying agent parameters, simulation designers can change how an agent interacts, what an agent does when it interacts, and with whom an agent chooses to interact. Though it is possible to give two agents identical parameter values at the beginning of the experiment, their patterns of interaction will be slightly different due to random factors. For this reason, it is important to note that parameters only suggest a general behavior pattern.

There are seven important parameter categories: an agent's *knowledge and beliefs*, its *internal characteristics*, its *task and performance characteristics*, its *message content parameters*, its *interaction characteristics*, its *interaction patterns*, and its *statistical parameters*. These categories are expanded in turn in the subsections below.

## 2.1 Knowledge & beliefs

Construct agents store multiple types of information, information used in making decisions, performing tasks, or communicating with others. Information can come in many flavors, from basic information about facts to meta-information about agents. This subsection describes the seven types of information that an agent can store: facts, beliefs, who-knows-who transactive memory, who-knows-what transactive memory, who-does-what transactive memory, beliefs transactive memory, and referrals.

- *Facts*: Facts represent the basic unit of information known by the agent. The number of facts, the meaning of the facts, and the interpretation of the facts is left to the experimenter; good experiment designers should carefully

consider these properties when designing an appropriate simulation. Once the meaning of the facts have been established, Construct agents can acquire facts in one of three ways.

  - An agent can interact with other agents to learn facts. When one agent learns a fact from another, the sender agent does not forget the fact; instead, at the conclusion of the interaction both the sender and receiver know it. The parameters that govern this type of knowledge transmission are further described in section 2.5.
  - An agent can acquire facts by performing one or multiple tasks. As an agent performs a task, it may be able to acquire facts that the experiment designer has associated with that task. For more information on the learning by doing process, see section 2.3.
  - An agent can also be pre-assigned facts at the beginning of the experiment. For more information on this is done, see appendix A.2.

- *Beliefs*: A belief represents an agent's agreement with a particular opinion. Though normal facts can be either known, partially known, or not known at all, agents either agree with, disagree with, or do not have an opinion about a certain belief based upon a belief threshold. Beliefs are partially based on facts, as agents can learn facts which sway their beliefs. However, beliefs are also affected by external factors. Beliefs can be changed in one of five ways.

  - An agent can interact with other agents to learn belief facts. Belief facts are facts which have a direct impact on a belief, and learning these facts will cause an agent's belief to strengthen, weaken, or switch. When one agent learns a belief fact from another, the sender agent does not forget the belief fact; instead, at the conclusion of the interaction, both the sender and receiver know it.
  - An agent can acquire belief facts by performing one or multiple tasks. As an agent performs a task, it may be able to acquire belief facts that the experiment designer has associated with that task. These belief facts, in turn, may be associated with a specific belief and will sway that belief.
  - An agent's beliefs are affected by it beliefs in the prior period and by its influenceability. When an agent learns a new belief fact, the agent does not immediately switch beliefs; instead, over time, agents gradually update its beliefs based upon all known information based upon its influenceability. The prior belief ensures a smooth transition from one belief to the other since rapid switching of beliefs is not commonly observed. For more information about influenceability, see section 2.2.
  - An agent's beliefs are also affected by the beliefs and influentialness of other agents in an agent's interaction sphere. All agents are affected by the beliefs of those in their interaction sphere, though more influential agents have greater effects than others. By this mechanism, it is possible for a very influenceable agent to hold a belief that is in opposition to the facts it possesses. For more information about influentialness and influenceability, see section 2.2.
  - An agent's beliefs can also be pre-assigned via belief facts, prior beliefs, or both at the beginning of the experiment. Assigning a belief fact will provide a basis for the agent's opinion, but may be transmitted during subsequent interactions. Prior beliefs may have more effect in some simulations than others, depending upon the strength of the initial belief, the beliefs held in the agent's interaction sphere, and the influenceability of the agent.

- *Who-knows-who transactive memory*: Who-knows-who transactive memory represents information about what an agent knows of the interaction patterns of other agents. This information is stored in a binary fashion, so the holder of the transactive memory either knows or does not know of the existence of the agent-agent link. Agents can acquire who-knows-who transactive memory in four ways.

  - An agent will update its who-knows-who transactive memory about itself if it meet another agent. Both the initiator and recipient will know that a link exists between them.
  - An agent will update its who-knows-who transactive memory about another agent if it meets another agent. Once an interaction has taken place, each agent will know that its interaction partner knows that agent. Thus, the initiator will know that the recipient knows about the link, and the recipient will know that the initiator knows about the link.

  - An agent can interact with other agents to learn who-knows-who transactive memory facts indirectly through interaction. When one agent learns a transactive memory fact from another, the sender agent does not forget the fact; instead, at the conclusion of the interaction, both the sender and receiver will know it.
  - An agent can also be pre-assigned transactive memory facts at the beginning of the experiment.

- *Who-knows-what transactive memory*: Who-knows-what transactive memory represents information about what an agent knows of the knowledge of other agents. This information is stored in a binary fashion, so the holder of the transactive memory either knows or does not know of the existence of the agent-fact link. Agents can acquire who-knows-what transactive memory in five ways.
  - An agent will update its who-knows-what transactive memory about another agent if it learns a fact from its interaction partner. Since the sender agent will knew the fact in order to teach the receiver agent, the recipient agent will be able to infer that the sender knew the fact.
  - An agent will update its who-knows-what transactive memory about another agent if it teaches a fact to an interaction partner. Since the recipient will then know the fact, the sender will be able to infer that the recipient agent heard the fact.
  - An agent can interact with other agents to learn who-knows-what transactive memory facts directly through interaction. Through this process, an agent can learn that a third party knows a particular fact.
  - An agent will update its who-knows-what transactive memory about itself if it learns a new fact from a task or another outside source. Since the agent knows a fact, it will update its transactive memory to indicate that it knows it knows a fact.
  - An agent can also be pre-assigned who-know-what transactive memory at the beginning of the experiment.

- *Who-does-what transactive memory*: Who-does-what transactive memory encode information about the task assignments of other agents. This information is stored in a binary fashion, so the holder of the transactive memory either knows or does not know of the existence of the agent-task link. Agents can acquire who-does-what transactive memory in three ways.
  - An agent will update its who-does-what transactive memory about itself if it performs a task. Since the agent performs a task, it will update its transactive memory to indicate its task knowledge appropriately.
  - An agent will update its who-does-what transactive memory about another agent if it performs a task in conjunction with another agent. Since the two interaction partners work together, each will be able to infer that the other performs the task.
  - An agent can also be pre-assigned who-does-what transactive memory at the beginning of the experiment.

- *Beliefs Transactive Memory*: Who-believes-what transactive memory represents information about the beliefs of other agents. This information is stored in a *ternary* fashion: the holder of transactive memory thinks the agent holds the belief, thinks the agent does not hold the belief, or does not know. Construct does not draw a distinction between "does not know" and "knows another agent has no belief" at this time, though this extension is under development as of this writing. Agents can acquire who-believes-what transactive memory in five ways.
  - An agent will update its who-believes-what transactive memory about itself if it changes its belief. Since the agent changed its belief, it will update its beliefs transactive memory about itself.
  - An agent will also update its who-believes-what transactive memory about another agent if it learns the belief of another agent while interacting. Since the sender agent explicitly passed a belief fact in a message, the recipient can update his who-believe-what transactive memory appropriately.
  - An agent will also update its who-believes-what transactive memory about another agent as it examines the beliefs in its interaction sphere. Since the agent must know the prevailing beliefs in its interaction sphere in order to do this, the agent in question will be able to update its appropriate who-believes-what transactive memory.
  - An agent can also be pre-assigned who-believes-what transactive memory at the beginning of the experiment.

- *Referrals*: Referrals are suggestions which may lead to a gain of information; agents who follow referrals are sent by a second party in search of a third. Referrals represent a goal-directed search in which an agent actively seeks an interaction partner who may be able to convey the agent a particular piece of information. Agents usually follow a referral to learn a particular fact, but they can also be referred to update a belief depending upon the simulation. For additional parameters which govern the referral process, see section 2.6

## 2.2 Internal characteristics

An agent's internal characteristics define an agent in the absence of an interaction partner. Many of these parameters do not affect the interaction process directly, but can affect pre- and post-interaction decisions. Important internal characteristics include: influentialness rate, influenceability rate, attention rate, forgetting rate, risk aversion rate, socio-demographic parameters, and physical proximity parameter.

- *Influentialness*: The influentialness parameter specifies how strongly an agent influences the beliefs of agents in its interaction sphere. Agents with higher influentialness will have a greater effect on the beliefs of the agents around them, while those with lower influentialness will have less. When spreading influence, an agent's influentialness is tempered by the other agent's influenceability parameter: even agents that are very influential will not affect the beliefs of agents with low influenceability. Note that while influentialness affects the overall beliefs of other agents, it has no effect on the communication of facts, belief facts, or any other form of knowledge. For more information about beliefs, see section 2.1.

- *Influenceability*: The influenceability parameter specifies how resistant an agent is to changing beliefs. Agents with higher influenceability are more susceptible to influence of those around them; agents with low influenceability can be thought of as stubborn and independent. Additionally, influenceability also affects how rapidly agents change their beliefs based on new information; agents with low influenceability will be reluctant to change their beliefs regardless of the facts they know. Note that while influenceability affects overall belief, it has no effect on the communication of facts, beliefs facts, or any other form of knowledge.

- *Attention*: The attention parameter affects how much information the agent examines when communicating with another agent. Attention manifests itself in two ways.

  - An agent's attention affects the amount of knowledge that it considers prior to communicating with another agent. An agent does not and cannot communicate everything it knows when interacting; instead, it focuses on a fraction specified by the attention rate. An agent with a high attention rate will examine more of what it knows when interacting, increasing the chance of a knowledge exchange. If the attention is set low enough, an agent may not send a message when communicating.
  - An agent's attention also affects the amount of information an agent receives from another agent. An agent with a high attention rate will retain more of what was communicated during an interaction, while an agent with low attention may fail to absorb new knowledge from an interaction partner. If attention is low enough, an agent may not record a message when communicating.

- *Forgetting*: The forgetting parameter affects how fast an agent loses information that it already knows. Agents can forget any type of information, even pre-assigned initial knowledge, unless that information is marked as unforgettable by the simulation designer. If a fact is not communicated during an interaction period, this parameter specifies its chance of being totally or partially forgotten. Since Construct facts can be represented using real numbers, it is possible for facts can be partially forgotten. In this way, it is possible for an agent to know part of a fact but not all of it.

- *Risk aversion*: The risk aversion parameter affects whether or not an agent can make a particular decision. Agents can accumulate beliefs during the simulation, as described in section 2.1, and in the absence of risk aversion will act upon these beliefs. The risk aversion parameter, however, acts as a catch-all for factors not included in the model which prevent an agent from acting on a particular belief. Agents who are risk averse will still be able to communicate their facts and beliefs like any other agent, but will never be able to make the corresponding decision.

- *Socio-demographics*: The socio-demographic parameters help define an agent's likely interaction partners. When selecting an interaction partner using homophily, as described in section 2.6, agents will attempt to chose a partner with similar or matching socio-demographic characteristics. Agents in many Construct simulations will include socio-demographic attributes for gender, age, education, and income, though others may include additional characteristics. Some agents, especially non-human agents, may be able to take on multiple socio-demographic values; this has the potential to increase homophily with different agents. In some experiments, agents may deemphasize the effect of socio-demographic parameters and choose to interact primarily based on knowledge; in others, especially in stylized experiments, socio-demographic characteristics may be ignored completely. This can be accomplished by modifying the relative socio-demographic weighting parameter for the entire experiment. For more information as to how socio-demographics are used to create derived parameters, see appendix A.1.

- *Physical proximity*: An agent's physical proximity to another agent has two interpretations, but both interpretations affect how likely it is that two agents will interact. For more information on how physical proximity is used in the simulation, see appendix A.1.

  - At a literal level, an agent's physical location and physical proximity represents the physical distance between one agent and another. Because Construct assumes that two agents that are farther away will interact less frequently, decreasing the physical proximity will make it less likely for two agents to communicate. This parameter may eventually be replaced by a more sophisticated geo-spatial modeling capability based on agent coordinates, though as of this writing this feature has not yet been implemented.
  - At a more abstract level, physical proximity serves as a catch-all interaction parameter. Physical proximity represents a convenient way of encoding other factors which can affect the interaction frequency between two agents. If one agent should have decreased interaction frequency with another, the inter-agent physical proximity could be set appropriately.

## 2.3 Task & performance characteristics

During the course of a simulation, Construct agents can perform particular tasks. These tasks may be the goal of the simulation, or they may serve as a way for agents to learn new information. While agents can perform tasks for any reason, their task assignment can have a significant effect on their later interactions. This subsection describes the three task-related parameters that an agent can have: task counts, learning by doing rate, and network type.

- *Task count*: Tasks represent activities that an agent may or must perform; the task count parameter specifies how many tasks an agent may or must perform. While the types of tasks must be specified by the experimenter, the simulation agent may make various choices as to when and how to accomplish the tasks. Tasks may require multiple interaction periods to complete depending upon the difficulty of the task, the granularity of the simulation, and additional design parameters. The total number of tasks that an agent can perform over the course of the simulation is currently unbounded, this may be changed in future versions.

  - *Binary tasks*: A binary task is a task in which agents compare their knowledge against a set of required bits of information. If the agent knows a particular fact and the fact is relevant for the completion of the binary task, the agent can use that fact; if an agent does not know the fact, it guesses. The task response is a compilation of an agent's knowledge and guesswork and is then compared to the ground truth specified by the experiment designer. This ground truth value is used to determine the task outcome and is used in calculating rewards or recording performance statistics. An agent can only work on one binary task at a time, and this binary task may take one or several periods to perform.
  - *Energy tasks*: An energy task is a task that require an agent to expends energy when performing. Because agents only have a limited amount of energy, they can only perform a certain amount of an energy task during any given period. If an agent has insufficient energy to perform a task, it must either wait another period to replenish its energy supply or give up and try a different task. When the agent has sufficient energy, it can attempt to complete the task and move on to the next one. An agent can only work on one energy task at a time, and this energy task may take one or several periods to perform.

- *Learning by doing*: The learning-by-doing parameter specifies how rapidly an agent can learn information, usually facts, when performing a particular task. If learning-by-doing is enabled for a particular simulation, an agent capable of learning-by-doing will have a small chance of learning a piece of information if the experimenter associates it with a particular task. Any information learned through the learning-by-doing process is treated identically to information learned via interaction; this information can be exchanged with other agents using the same exchange mechanism, can be kept for use in making decisions, or can be forgotten. Agents may have different learning-by-doing rates for different tasks or different facts, since it may be easier for agents to learn facts from some tasks than from others.

- *Network type*: The network type parameter specifies the type of network with which an agent is associated, either a communication network or a command network. These parameters specify how task performance information flows between agents.

  - An agent with a communication network has a two-way network in which information exchange occurs reciprocally. When communication occurs in a communication network, the initiator and recipient of communication can each send information to the other. Agents in a communication can communicate reciprocally, so two agents in a communication network can both gain information from interacting.
  - An agent in a command network has a directed network, either hierarchical or matrix in nature. When communication occurs in a command network, either the initiator or the recipient sends information, but not both. Only one agent in a command network can gain information from an interaction in a given period, so either one agent will give orders down or another agent will send information up. Agents in a command network have strict rules on what and how they report, and they may have other limitations as well depending upon further simulation parameters.

## 2.4 Message content parameters

The message content parameters specify the content types that an agent can transmit during an interaction. While agents might be able to represent certain pieces of information internally, as described in section 2.1, they may not be able to communicate them. There are seven types of information that can be included in a message: facts, beliefs, who-knows-who transactive memory, who-knows-what transactive memory, who-does-what transactive memory, who-believes-what transactive memory, and referrals.

- *Transmits facts*: The transmits facts parameter specifies whether an agent is capable of sending or receiving facts in a message. It may be possible for an agent to learn facts without being able to transmit them, for instance by performing tasks. Agents need not be capable of both sending and receiving facts, as some agents can send without receiving while others may be able to receive without sending.

- *Transmits beliefs*: The transmits beliefs parameter specifies whether an agent is capable of sending or receiving beliefs in a message. It may be possible for an agent to learn beliefs without being able to transmit them, for instance by inferring them from their own knowledge. Agents need not be capable of both sending and receiving beliefs, as some agents can send without receiving while others may be able to receive without sending.

- *Transmits who-knows-who transactive memory*: The transmits who-knows-who transactive memory parameter specifies whether an agent is capable of sending or receiving who-knows-who transactive memory in a message. It may be possible for an agent to learn this kind of transactive memory independently without being able to transmit them, for instance by interacting with other agents. Agents need not be capable of both sending and receiving who-knows-who transactive memory, as some agents can send without receiving while others may be able to receive without sending.

- *Transmits who-knows-what transactive memory*: The transmits who-knows-what transactive memory parameter specifies whether an agent is capable of sending or receiving who-knows-what transactive memory in a message. It may be possible for an agent to learn this kind of transactive memory independently without being able to transmit them, for instance by learning a fact from another agent and updating the transactive memory for the sender agent. Agents need not be capable of both sending and receiving who-knows-what transactive memory, as some agents can send without receiving while others may be able to receive without sending.

- *Transmits who-does-what transactive memory*: The transmits who-does-what transactive memory parameter specifies whether an agent is capable of sending or receiving who-does-what transactive memory in a message. It may be possible for an agent to learn this kind of transactive memory independently without being able to transmit them, for instance by working jointly on a task with another agent. Agents need not be capable of both sending and receiving who-does-what transactive memory, as some agents can send without receiving while others may be able to receive without sending.
- *Transmits who-believes-what*: The transmits who-believes-what transactive memory parameter specifies whether an agent is capable of sending or receiving who-believes-what transactive memory in a message. It may be possible for an agent to learn this kind of transactive memory independently without being able to transmit them, for instance by examining the beliefs in the interaction sphere. Agents need not be capable of both sending and receiving who-believes-what transactive memory, as some agents can send without receiving while others may be able to receive without sending.
- *Transmits referrals*: The transmits referrals parameter specifies whether an agent is capable of sending or receiving referrals in a message. Agents need not be capable of both sending and receiving referrals, as some agents can sending without acting upon them while others may be able to act upon referrals without sending them. For more information on referrals, see section 2.1.

## 2.5 Interaction characteristics

The interaction characteristics specify how often an agent will interact with other agents and what it will transmit when it interacts. While other agent parameters govern the high-level details of interaction partner selection, the interaction characteristics specify low-level parameters that govern how much interaction is exchanged. This subsection describes the two important interaction characteristics: interaction parameters and message parameters.

- *Interactions*: The interactions parameter specifies the combined number of interactions that an agent can have during a given time period. When interacting, agents can either initiate communication, seeking out an interaction partner of choice, or respond to communication, waiting for an agent to initiate communication with them. While there is no current limit on the total number of interactions an agent can have during a single interaction period, future versions of the tool may limit this.
  - *Initiations*: An agent's initiation parameter specifies the number of other agents that the agent can choose to contact each round. At the beginning of a round, all agents capable of initiating communication are ordered randomly; during the period, agents are pulled from this queue and allowed to select any available interaction partner for as many times as they can initiate interaction. Agents cannot initiate communication with more agents than their maximum initiation count; however, if they are at the bottom of the queue and there are no agents who can receive communication, then it is possible for agents to initiate communication with fewer than their minimum amount.
  - *Receptions*: An agent's reception parameter specifies the number of other agents that can contact the agent each round. Agents are selected for reception based upon another agent's interaction parameters; the reception partner waits in order to be contacted. Agents cannot receive communication from more agents than their maximum reception count, though they may receive fewer communications than their minimum amount if there are insufficient initiators. Though communication reception is a passive role in the sense that the recipient does not actively seek out an initiator, both initiators and recipients will transmit facts once the interaction begins.
  - *Unique interaction partners*: An agent's unique interaction partners parameter specifies the number of different interaction partners that an agent may have during a single time period. If the number of unique interaction partners is less than the total number of interactions, it is possible for an agent to interact with the same agent multiple times in a given time period. This repeated interaction represents an extended communication period between two agents, a process that allows two agents can exchange much more information than would otherwise be possible.
- *Message*: The message set of interaction characteristic parameters govern the amount of information that is sent between two interacting agents. When agents interact, they exchange a message – a collection of information –

which they use to update their stored knowledge. The complexity of a message sent between two agents is highly dependent on the fidelity of simulation being modeled. In some models, complex messages will be represented using long messages; in others, complex message will be represented using short ones. In any well-designed simulation, however, agents who send simple messages will send shorter messages than those who send complex ones regardless of the simulation fidelity.

  - *Message length*: An agent's message length specifies the number of pieces of information that can be sent in a message. At this time, it is not possible to separately limit the amount of facts, beliefs, or transactive memory exchanged during a given interaction; the message length parameter limits only the total amount of information exchanged, not the individual possible components in the message. For more information as to what information can be included in the message, see section 2.4.
  - *Distinct messages*: An agent's distinct message count parameter specifies the number of different messages an agent can send each period. Agents that interact multiple times may communicate the same message to multiple other agents or may have the ability to send different messages to different interaction partners. If an agent sends two or more distinct messages, then the process of choosing the information each message is independent of the other choice process. It is important to note that the number of distinct messages sent is independent of the number of distinct messages received: agents can receive the same sent message differently due to differences in attention, though these messages do not count against the sender's distinct message count. If an agent can only communicate with one other agent, than this parameter is ignored.

## 2.6 Interaction patterns

The interaction patters parameter specifies a variety of high-level parameters related to how an agent chooses an interaction partner. Generally, these parameters serve to narrow down the interaction partner specification procedure and greatly improve Construct runtime performance. There are three major interaction patterns parameters: interaction sphere parameters, agent isolation parameters, and interaction strategy parameters.

- *Interaction sphere*: An agent's interaction sphere defines the number of agents with whom an agent can interact, as agents can only interact with agents in their interaction sphere. If one agent is not in another's agent interaction sphere, the first will not be able to interact with the second directly though they may be able to exchange information by a series of third-party intermediates who link the two spheres. Interaction spheres need not be symmetric.
  - *Sphere density*: An agent's sphere density parameter specifies the proportion of agents who can appear in a given agent's interaction sphere. The sphere density is a percentage, not a constant number, so simulators who want a constant-sized interaction sphere should consider adjusting the sphere density if the number of simulated agents is modified.
  - *Sphere generation mechanism*: An agent's sphere generation mechanism parameter specifies how its interaction sphere is built. One way to create an interaction sphere is to have the experimenter generate the sphere externally in a CSV file and supply it to Construct. Interaction spheres built in this way will not vary between runs and are best if the spheres should not vary between simulation runs. Additionally, Construct provides two mechanisms for building the sphere internally: social-demographic similarity matching and two-away matching. Social-demographic similarity matching builds an interaction sphere by giving preference to agents who have similar socio-demographic characteristics. These interaction sphere generators include some random elements as tiebreakers, so the interaction spheres will not be identical between simulation runs. Two-away matching builds a sphere by generating a graph and, for each agent, letting the interaction sphere equal all the agents in that agent's initial size-two ego net. In order to build these interaction spheres, it is necessary to use a command network in the experiment or otherwise supply the network hierarchy. For more information about sphere generation mechanisms, see appendix B.
  - *Sphere modifications*: An agent's sphere modification parameter specifies if the interaction sphere can be modified during the course of the experiment. Though the current implementation of Construct relies on a static interaction sphere, future versions may allow the sphere to change. If the sphere size changes, then the number of agents in the interaction sphere would be able to increase or decrease over time, shrinking

or expanding an agent's pool of potential interaction partners. Alternatively, the sphere composition may be able to change and allow agents currently in the initial pool to be replaced by others outside it.

- *Agent isolations*: The agent isolation parameter specifies whether or when an agent can be suspended from the simulation. When an agent is isolated, it is ignored for the interaction period: it cannot initiate or receive communication, update its beliefs, perform tasks, or make decisions. Isolations can last one or more periods, can occur repeatedly, and can be cyclic. Isolation is a commonly used tactic for non-human agents, as the effects of non-human agents is generally intended to occur infrequently. For more information about isolations, see appendix B.

- *Interaction strategy*: The agent interaction strategy parameter specifies the process that an agent uses to choose an interaction process during an interaction round. In the current implementation of Construct, the experimenter must pre-specify the percentage of the time that an agent uses each of these three interaction patterns; in future versions, it may be possible for these percentages to change over the course of the experiment. Agents can interact in one of three ways.

  - An agent that acts via homophily attempts to find an interaction partner that shares its characteristics. When searching for suitable partners, the agent will stress agents who have similar socio-demographic parameters, similar knowledge, and similar beliefs. This process utilizes the derived parameters described in appendix A to find the most suitable interaction partner or partners.
  - An agent that acts via deliberate search will attempt to find an interaction partner that knows a particular piece of information. When searching for suitable partners, the agent will stress knowledge and will ignore most other parameters. Agents who lack the piece of information will be ignored by the seeker.
  - An agent that acts with its co-workers will interact with those agents that are performing a similar task. When searching for interaction partners, the agent will stress tasks primarily and will ignore most other parameters. Agents assigned to other tasks will be ignored by the seeker.

### 2.7 Statistical parameters

In addition to the parameters mentioned in previous sections, there are two parameters which are useful for analysis of simulation results. While these parameters have no effect on the simulation while it is running, they can be extremely valuable for post-hoc analysis.

- *Agent name*: The agent name serves as a marker for the agent in the simulation. The agent name does not affect the agent's performance in the Construct simulation itself, though unique agent names can be helpful for subsequent data analysis. If no agent name is specified when the agent is created, the agent can be located through a unique, system-assigned agent ID.

- *Agent groups*: Agent groups help to collect data on a number of agents simultaneously in order to facilitate post-experiment statistical analysis. In past implementations of Construct, agents could only be members of one group at a time; however, newer versions of Construct will relax this restriction in order facilitate statistics gathering. While it may make sense to have groups that are homogeneous over a specific parameter, such as all agents with a particular socio-demographic characteristic, group composition is left to the agent designer.

## 3 Agent classes

An *agent class* is a template for creating agents. Agent classes are distinct from agent groups: while agent classes specify agent information, agent groups are used for statistic-gathering purposes only. Using agent classes can help simplify the experimental design procedure for two important reasons. First, agent classes can be designed such that all agents of that class will have identical values for certain properties. This will allow the experiment designer to set the property once per agent class and to minimize the number of changes should the value need to be modified. Additionally, agent classes can be used to specify a range for a specific parameter. While these ranges serve as hard caps on the parameter values, Construct can choose a random value in that range in order to provide an additional

Table 1: Human classes

| Characteristic | Human | + TM | + Beliefs | + Mass Media |
|---|---|---|---|---|
| Communication Style | one-to-one | one-to-one | one-to-one | one-to-many |
| Initiates Communication | yes | yes | yes | yes |
| Receives Communication | yes | yes | yes | yes |
| Message Complexity | simple | simple | simple | simple |
| Learns Knowledge | yes | yes | yes | yes |
| Forgets Knowledge | yes | yes | yes | yes |
| Has Transactive Memory | no | yes | no | no |
| Has Beliefs | no | no | yes | no |
| Has Default Knowledge | high | high | high | moderate |
| Is Isolated | never | never | never | rarely |

Table 2: Media classes

| Characteristic | Book | Leaflet | Flier | Web site |
|---|---|---|---|---|
| Communication Style | one-to-many | one-to-many | one-to-many | one-to-many |
| Initiates Communication | no | no | no | no |
| Receives Communication | yes | yes | yes | yes |
| Message Complexity | complex | simple | simple | complex |
| Learns Knowledge | no | no | no | no |
| Forgets Knowledge | no | no | no | no |
| Has Transactive Memory | no | no | no | no |
| Has Beliefs | no | no | yes | yes |
| Has Default Knowledge | moderate | low | low | low |
| Is Isolated | rarely | very often | very often | rarely |

Table 3: Database classes

| Characteristic | Task | Referential | Avatar |
|---|---|---|---|
| Communication Style | one-to-many | one-to-many | one-to-many |
| Initiates Communication | no | no | no |
| Receives Communication | yes | yes | yes |
| Message Complexity | complex | complex | simple |
| Learns Knowledge | yes | yes | no |
| Forgets Knowledge | no | no | no |
| Has Transactive Memory | no | yes | no |
| Has Beliefs | no | no | no |
| Has Default Knowledge | low | low | moderate |
| Is Isolated | rarely | rarely | rarely |

Table 4: Financial education classes

| Characteristic | Promoter | Seminar | Advertisement | Opinion Leader |
|---|---|---|---|---|
| Communication Style | one-to-one | one-to-many | one-to-many | one-to-many |
| Initiates Communication | yes | no | yes | yes |
| Receives Communication | yes | yes | yes | yes |
| Message Complexity | simple | complex | simple | simple |
| Learns Knowledge | no | no | no | no |
| Forgets Knowledge | no | no | no | no |
| Has Transactive Memory | yes | no | no | yes |
| Has Beliefs | yes | yes | yes | yes |
| Has Default Knowledge | moderate | moderate | low | moderate |
| Is Isolated | rarely | often | very often | often |

random element to the simulation. Either of these mechanisms can be used to set any of the per-agent parameters described in section 2.

Past Construct simulations have generally worked with many different types of agent classes, including *human classes*, *media classes*, *database classes*, and *financial resource classes*. These are broad class categories, and there is substantial variety within each of them. By associating classes in this way, however, it is easier to understand the differences between the individual classes and to determine which type of class is most appropriate for a given simulation. The remainder of this section will discuss these class categories and discuss various texts which may provide additional information about their use. For more information about a particular agent class, appendix C contains detailed style sheets which describe the Construct parameters set in agents of that class.

## 3.1 Human classes

Human classes, detailed in table 1, consist of standard human agents [3]. Human agents represent people who can interact, share knowledge, and perform tasks during a virtual experiment. Depending upon the type of virtual experiment being modeled, agents will need additional complexity such as beliefs or transactive memory. For results from past experiments using human agents, see [7, 8, 4].

There are a variety of human classes, some of which are more complex than others. There are four different types of human classes, each designed to be used in different virtual experiments: the *simple human* class, the *human with transactive memory* class, the *human with beliefs* class, and the *human with mass media* class. The simple human class is the simplest form of agent possible, as it lacks transactive memory and beliefs, and can be best for low-granularity simulations with tight memory requirements. The humans with transactive memory are much better representations of humans, but may require additional resources to use in large numbers. Humans with beliefs will use belief facts during simulations and will take actions based on those beliefs. Humans with access to mass media will be able to send information to multiple agents in a single interaction period. All of these human agents can both initiate and receive communication, learn and forget information, and are generally active for the entire simulation. Appendix C specifies a number of additional human agent classes that share some combination of these properties.

## 3.2 Media classes

Agents of the media classes, detailed in table 2, are agents that store information in some form of media [1, 3]. Media classes can store multiple facts and pass them to other agents in the simulation, usually interacting with multiple agents at once to send a message. However, these media classes cannot learn new information and do not store transactive memory about other agents. Past results from virtual experiments using media classes have indicated that information tends to travel much more rapidly when these agents are present, though this additional information may not always lead to improved task performance [4, 9]. For results from past experiments using media agents, see [4].

There are four major media classes. *Books* are large repositories of knowledge intended to be read by only a few people at a time. *Leaflets* are meant to convey a small amount of information to a large number of people. *Fliers* are meant to convey beliefs to a large number of people, and are slightly more powerful than leaflets. Lastly, *web sites* are meant to convey custom amounts of information and beliefs to large numbers of people. All media agents can receive communication from multiple agents, as it is assumed that access to these agents is non-exclusive during an interaction period. Media agents cannot initiate communication, but when they receive communication they can transmit a complex message. Media agents are static once created, meaning that they cannot learn, forget, or develop transactive memory. Certain media agents may carry associated beliefs, though the meaning of these beliefs should be specified by the simulation designer. In some virtual experiments, it may be logical to have media agents available for all time periods, while in others, it may make more sense for these agents to be available only intermittently.

## 3.3 Database classes

Agents of the database classes, described in table 3, are agents designed to assist others in performing a particular task [4]. Unlike traditional media classes, databases can increase their knowledge over the course of the virtual experiment. Because databases can be accessed by many others simultaneously, it is possible for them to increase the rate at which information diffuses through an organization or network. However, some Construct results have indicated that blindly using such technology may have negative effects on overall task performance [9]. For results from past experiments using database agents, see [4, 9].

There are three major database classes. *Task databases* store facts usually facts relevant to the completion of a particular task. *Relational databases* store transactive memory information, though some store task information as well. *Avatars* store small amounts of information for a human agent. All of these agent classes can interact with multiple agents simultaneously, but they do initiate communication with human agents. These agents can generally transmit complex messages, learn new knowledge, and retain all knowledge that they have learned. By default, they do not have beliefs associated with them, though they may affect the beliefs of their interaction partners by teaching them new belief facts. Generally, these agents are active for the entirety of the virtual experiment, and many begin with limited knowledge.

### 3.4 Financial education classes

Agents of the financial education classes, described in table 4, are an eclectic group of agent classes with particular relevance to financial simulations [10]. These agents generally spread facts and beliefs associated with a certain financial action, though the exact nature of their messages should be customized by the experiment designer. For results from past financial education class experiments, see [6, 10].

There are four major financial agent classes. Agents of the *promoter* class are human agents with fixed beliefs about the legality of a particular scheme. Though they act like a standard human agent in many respects, their beliefs and knowledge are pre-specified and do not change over the course of the simulation. Agents of the *seminar* class provide in-depth information about a specific financial strategy. The seminar agent class is similar to be the book class described earlier, except for the fact that the seminar class has specific user-defined knowledge, can convey user-specified beliefs. Agents of the *advertisement* class provide rudimentary information about a financial strategy; they are similar to agents of the flier class described earlier but have pre-specified beliefs. Agents of the *opinion leader* class are important members of society, such as mayors and public officials. Though they are similar to human agents with mass media, their beliefs must be pre-set by the simulation designer. Agents of the financial education classes generally do not learn or forget knowledge between interactions, as they tend to be used as interventions on an human social network. Additionally, agents of these classes are rarely active; in most virtual experiments, they are usually present for only a handful of periods.

### 3.5 Custom classes

Often, simulations will require more specific or more complex agents than the default ones specified above. Some of these can be created with only a minimum amount of work by extending one of the above classes and modifying one or two parameters. In this case, the new classes will have similar characteristics to the ones specified above. Alternatively, agent classes can be created from scratch, a process which provides maximum freedom to the simulation designer. While this is a more time-intensive process, it will give the simulation designer a better understanding of the agent class and its capabilities. Advanced experiment designers who choose to create their own classes should refer to appendix B for further information on agent class parameters and class style sheet creation.

## 4 Conclusion

As simulation becomes more common in social network research, tools such as Construct are becoming increasingly important. To satisfy the need for more complex simulations, Construct has become more powerful. In the process, however, Construct has also become more complex and potentially more difficult to use. This technical report has described the parameters necessary to specify agents in Construct by explaining how to specify agent parameters, knowledge, beliefs, and classes. It has also summarized a number of default agent classes commonly used with the simulation system. Users who are familiar with the settings described in this report will have a richer understanding of Construct's capabilities, and they will be better prepared to design successful virtual experiments.

Because virtual experiments and their interventions are becoming more and more elaborate, it has become necessary to develop a graphical user interface to assist the setup procedure. Though there have been several past efforts to develop a Construct general interface [2, 11], this work was under-appreciated and not widely used. In January 2007, work began on a graphical user interface for Construct, and as of this publication the tool is still under development. This forthcoming GUI interface takes inspiration from other successful interfaces from the CASOS lab, such as those for ORA [12] and AutoMap [13], and its format should be familiar to those who have worked with CASOS tools in

the past. It is widely expected that this interface will greatly facilitate the creation, modification, and maintenance of Construct agents, and will greatly simplify the agent specification process.

# Appendix

# A Additional Information

This section provides supplemental information about how Construct uses and manipulates agent parameters, specifically in terms of derived parameters and element generators.

## A.1 Derived parameters

While the user has the ability to specify a number of parameters for the simulation, it should be noted that these parameters are use indirectly to change agent behavior. A variety of parameters are built from the run-time information available during the simulation, and these derived parameters are used to manipulate agents. Though these parameters are heavily influenced by the selection of values for the parameters of section 2, the parameter values for other agents and random values may have a significant effect on agent behavior.

- *Social proximity*: Social proximity is a parameter used when selecting an interaction partner. Social proximity is primarily affected by socio-demographics, physical proximity, and knowledge similarity. The closer two agents are in each of these areas, the more likely they will be to interact. Social proximity is also affected by various weighting measures which affect the simulation globally, so each of the components can be given more or less weight.
- *Expertise*: An agent's expertise is a function of the knowledge that it knows at a point in the simulation, and helps an agent choose an interaction partner. Expertise can increase over time as agents learn new knowledge from tasks or via interacting, but it can also diminish if knowledge is forgotten. Expertise is primarily a function of knowledge, particularly knowledge associated with a particular set of facts. It is important to note that expertise is not the same thing as influentialness: an agent may be an expert without being very influential, and an agent can be very influential without being very informed.
- *Relative expertise*: An agent's relative expertise is a function of its expertise and the expertise of an agent with which it is interacting, and helps affect the transfer of knowledge between agents. While an agent may be an expert in one interaction, it may not be an expert in another interaction since relative expertise partially depends upon the knowledge of the interaction partner. In general, agents prefer to seek out others who are relative experts, especially when attempting to accomplish a particular task.
- *Expertise influence*: Expertise influence represents the influence of an informed opinion holder and can affect the mechanism by which an agent communicates beliefs. Expertise influence is generally a function of knowledge, though it can be affected by other factors. If an agent's interaction partner is an expert in a certain area, it may affect the agent's beliefs by virtue of the fact that the agent is an expert. Note again that expertise influence is not the same thing as influence; an agent can have a high expertise influence without being very influential overall.
- *Belief conflict*: Belief conflict occurs when an agent makes a decision based on two incompatible beliefs. For instance, an agent may believe that an action is illegal but that it is also a good thing to do; such an impasse is resolved by a general simulation rule to let one belief dominate the other. Belief conflict is affected by beliefs, agent facts, and the risk aversion rate, but ultimately must be resolved by good simulation design.

## A.2 Generators

Generators provide a convenient method for including a random element in a simulation, and as such can be very useful to tweak runs once all the other parameters have been specified. Some generators rely on a random number generator, so if a different random seed is used then a different generator outcome will occur. Construct includes several types of generators which can be used to set parameter values, including the *constant generator*, the *random uniform generator*, and the *random binary generator*. Alternatively, Construct allows a generator to read input values from external files via the *CSV file generator* or *DynetML file generator*

- *Constant generator*: The constant generator is used to set constant values. Generally, the constant value generator is used to set binary values, but where useful it can be used to set floating-point number constants. The constant generator is used in three main areas. First, constant generators can be used to explicitly set the value of a single parameter to ensure an agent has a particular fact. Second, it serves as a useful shorthand for setting multiple parameters to the same value – for instance, to ensure that a particular agent knows every fact. Last, it can be used to initialize all values to zero; values are not initialized to zero on default, and if a number is not explicitly set to zero (or to another value) then its value is system-dependent.

- *Random uniform generator*: The random uniform generator is used to set a value to a random double-precision value between a minimum and a maximum. If multiple parameter values are set using a random uniform generator, then each random value is chosen independently using the same minimum and maximum. Random uniform generators are useful for specifying non-binary information.

- *Random binary generator*: The random binary generator is used to set a bit based on a random value. For a random binary generator, a threshold between zero and one must be specified; the random generator will then choose a number between zero and one and will set the bit if the random number is less than the threshold. If multiple parameter values are set using a random binary generator, then each random value is chosen independently. Random binary generators are useful for information for which fractions do not make sense, such as whether an agent is in another's interaction sphere or if an agent is active in a specified time period.

- *CSV file generator*: The CSV file generator is used to initialize a parameter using the values stored in a particular CSV file. Generally, this type of generator should be used in cases where values should stay constant over multiple runs but where it might be easier to modify values in a CSV file than re-coding them using generators.

- *DynetML file generator*: The DynetML file generator is used to initialize a parameter using the values stored in a particular DynetML file. This type of generator should be used if it would be helpful to manipulate the generator data using another CASOS tool such as ORA.

While generators are most commonly used is to specify initial values for agent knowledge or beliefs, they can also be used in other places. For instance, generators can be used to specify the information that an agent is not allowed to know, that an agent is required to always know, or even the agents that compose another agent's interaction sphere. If a non-standard generator is used to generate information, this should be specified in the agent class style sheet.

# B Class Specification Style Sheets

A class specification style sheet documents the important features of an agent class. It provides a consistent way of documenting important fields of the class, allowing classes to be easily compared and understood. Agent class style sheets are two pages long: the first page, or *class overview page*, provides a short description of a class's overall behavior and documents necessary parameters; the second page, the *class details page* provides additional fields for free-form comments and additional information. A sample class style sheet is provided as table B

**Sample Agent Style Sheet (Overview Page)**

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | Yes |
| Has who-is-doing-what TM | No |
| Has beliefs TM | Yes |
| Has initial values | No |
| | |
| Follows referral suggestions | Yes |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | Yes |
| | |
| Can send beliefs | Yes |
| Can receive | Yes |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | Yes |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | Yes |
| Sends and receives | Yes |
| | |
| Can send referral | Yes |
| Can receive | Yes |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 1 |
| Maximum | 1 |
| Minimum receptions / period | 1 |
| Maximum | 1 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 1 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | ++ |
| Maximum in total | ++ |
| | |
| Performs binary tasks | Yes |
| Performs energy tasks | Yes |
| | |
| Can be pre-assigned to tasks | Yes |
| Tasks change over time | Yes |
| Can choose task assignments | Yes |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 1.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | Yes |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Sample Agent Style Sheet (Details Page)**

| **Default Knowledge** |
| --- |
| Additional information about default knowledge, or distribution of default knowledge among agents of this class, should be included here. |

| **Default Beliefs** |
| --- |
| Information about default beliefs, or distribution of default beliefs among agents of this class, should be included here. |

| **Active Periods** |
| --- |
| Information about isolations, and the nature and periodicity of isolations, should be included here. |

| **Interaction Notes** |
| --- |
| Information about interactions, such as agents which must not be included in an interaction sphere or required communication partners, should be included here. |

| **Internal Characteristic Notes** |
| --- |
| Information about internal characteristics, such as location or physical proximity, should be included here. |

| **Task Notes** |
| --- |
| Task information, such as specific task designations, should be specified here. |

| **Notes** |
| --- |
| This field provides additional information not covered in any of the above fields. |

## B.1 Class overview page

The first page of the agent class style sheet is divided into six tables. The three tables on the left-hand side of the page, the *knowledge and beliefs* table, the *internal characteristics* table, and the *tasks and assignments* table, describe the internal properties of the agent class. These parameters govern the kinds of information to which an agent of this class will have access, and will indirectly govern the types of agents with which an agent may interact. The three on the right hand side of the page, the *message content* table, the *interaction characteristics* table, and the *interaction patterns* table, describe interaction parameters of the agent. These parameters explicitly specify the numbers, types, and qualities of interaction between this agent and other agents in the simulation.

The knowledge and beliefs table specifies whether agents of this class can store the specified types of information. It also specifies whether agents have pre-set information when the simulation begins.

- *Has fact knowledge*: This parameter specifies whether agents of this class can store fact knowledge. If agents of a class have initial facts, then the facts (or the percentage of facts known) should be specified in the default knowledge field of the class details page.
- *Has beliefs*: This parameter specifies whether agents of this class can store beliefs. If agents of a class have initial beliefs, these beliefs should be specified in the default beliefs field of the class details page.
- *Has who-knows-who transactive memory*: This parameter specifies whether agents of this class have who-knows-who transactive memory and can keep track of agent-to-agent relationships. Giving an agent who-knows-who transactive memory will dramatically increase its memory requirements, since who-knows-who transactive memory requires an additional agent-by-fact matrix for each agent.
- *Has who-knows-what transactive memory*: This parameter specifies whether agents of this class have who-knows-what transactive memory and can keep track of agent-to-knowledge relationships. Giving an agent who-knows-what transactive memory will dramatically increase its memory requirements, since who-knows-what transactive memory requires an additional agent-by-fact matrix for each agent.
- *Has who-does-what transactive memory*: This parameter specifies whether agents of this class have who-does-what transactive memory and can keep track of agent-to-task relationships. Giving an agent who-does-what transactive memory will dramatically increase its memory requirements, since who-does-what transactive memory requires an additional agent-by-task matrix for each agent.
- *Has beliefs transactive memory*: This parameter specifies whether agents of this class have beliefs transactive memory and can keep track of what other agents believe. Giving an agent who-believes-what transactive memory will dramatically increase its memory requirements, since who-believes-what transactive memory requires an additional agent-by-beliefs matrix for each agent.
- *Has initial transactive memory values*: This parameter specifies whether the agent begins the simulation knowing any transactive memory values. Agents may begin with specific transactive memory values, or they may begin with some percentage of all transactive memory values. If the agent has any initial transactive memory, then this value should be set and the initial values for the particular matrix should be specified in the default knowledge field of the class details page.
- *Follows referrals*: This parameter specifies whether an agent will follow referral suggestions. If the agent has initial referral information, then that information should be specified in the default knowledge field of the class details page.

The internal characteristics table specifies a number of agent class parameters that affect how class agents internalize and process information. These parameters will have an indirect effect but substantial impact on how an agent of this class will choose to interact in a virtual experiment.

- *Influentialness rate*: This parameter, specified as a range with a min and a max, specifies how influential an agent of this class will be. Influentialness rates close to zero will mean that the agent will not have much sway over other agents, while values closer to one will mean that the agent has a significant amount of influence.

- *Influenceability rate*: This parameter, specified as a range with a min and a max, specifies how susceptible an agent of this class will be to the influentialness of another agent. Influenceability values close to zero will mean that an agent is extremely independent and unlikely to change its beliefs, while values close to one indicate that an agent is very impressionable.

- *Attention rate*: This parameter, specified as a range with a min and a max, determines how much information is transmitted and recorded when this agent interacts. Attention rates closer to zero will mean that the agent will consider less information when preparing a message and will be less likely to store information that it receives; rates closer to one will mean that the agent will consider more information when sending and will be more likely to store information when receiving.

- *Forgetting rate*: This parameter, specified as a range with a min and a max, specifies how fast an agent of this class will forget facts that it already knows. Forgetting rates closer to zero will mean that the agent will forget fewer pieces of information each period, while forgetting rates closer to one will lead to high forgetfulness. Agents that cannot forget should have a minimum and maximum forgetting rate of zero. If the forgetting rate must be customized for different facts, or if certain facts should never be forgotten, the minimum and maximum values should be set to ++ and details should be provided in the internal characteristics notes field on the class details page.

- *Risk aversion rate*: This parameter, specified as a range with a min and a max, specifies what percentage of agents of this class will be able to make a particular decision. Agents with risk aversion rates closer to zero will be more likely to be risk averse and to be unable to make a particular decision, while agents with risk aversion rates closer to one will be more likely to be able to act upon their fact and belief knowledge. It is important to note that this parameter defines a distribution over multiple agents of this class; each particular agent will or will not be risk averse but cannot be fractionally risk averse.

- *Has socio-demographic characteristics*: This parameter, if set, specifies that agents of this class have socio-demographic characteristics. These characteristics help to determine homophily when an agent chooses its interaction partners. Typical virtual experiments will have gender, age, income level, and education level characteristics, but other experiments may include race, number of children, and a variety of other customizable parameters as well. If agents of a class have required or forbidden characteristics, these features should be listed in the internal characteristic notes field on the class details page.

- *Has location*: This parameter, if set, specifies that agents of this class have a physical location. If the value should be decided randomly, this value should be set to yes. If the agent location is pre-set to a specific location, then the values should be set to ++ and details should be provided in the internal characteristic notes field on the class details page.

The tasks and assignments table specifies agent class parameters that deal with agent tasks. Tasks provide an alternative mechanism by which agents can gain information and also provide a convenient mechanism for bringing agents together.

- *Minimum number of tasks*: This parameter specifies the number of tasks an agent may perform at one time. If an agent is not able to perform any tasks, the minimum and maximum number should be set to zero. If the agent must perform a specific task, or must start out performing a specific task, this value should be set to ++ and details should be provided in the task notes field of the class details page.

- *Maximum number of tasks performed simultaneously*: This parameter specifies the maximum number of tasks that an agent of this class can perform during one interaction period. Currently, the maximum number of tasks that any Construct agent can perform simultaneously is fixed at one; agents that cannot perform tasks should have this value set to zero.

- *Maximum in total*: This parameter specifies the maximum number of tasks that an agent of this class can perform over the course of the virtual experiment. This value is a maximum value, so agents may not be able to complete this number of tasks due to time or other constraints. If the total number of tasks is unlimited, the value should be set to ++. This limit is currently not enforced and is included for forward compatibility.

- *Performs Binary Tasks*: This parameter specifies whether an agent can perform binary tasks. If the agent must perform specific binary tasks, this value should be set to ++ and details should be provided in the task notes field of the class details page.
- *Performs Energy Tasks*: This parameter specifies whether an agent can perform energy tasks. If the agent must perform specific energy tasks, this value should be set to ++ and details should be provided in the task notes field of the class details page.
- *Can be pre-assigned to tasks*: If this parameter is set, then agents of this class can be pre-assigned to a set of tasks. If agents of this class must be pre-assigned to specific tasks, then the value of this parameter should be set to ++ and the tasks should be enumerated in the task notes field of the class details page.
- *Task assignments can change over time*: If this parameter is set, then agents of this class will be able to change tasks. If it is unset, then agents will have fixed task assignments and will perform their tasks in a specified order.
- *Can choose task assignments*: If this parameter is set, then agents of this class will choose their own tasks to perform. Agents will use a number of criteria when deciding which task to perform, including knowledge and task requirements, beliefs, and expertise. If agents of this class cannot choose particular tasks, then this parameter should be set to ++ and the excluded tasks should be specified in the task notes field of the class details page.
- *Learning-by-doing rate*: This parameter, specified as a range with a min and a max, specifies how fast an agent will learn new facts from performing a task. An agent that cannot learn from its tasks should have a maximum interaction value of zero, while an agent who will be guaranteed to learn a fact should have a value of one.
- *Learning-by-doing rate is different across facts*: If this parameter is set, then the learning-by-doing rate is not constant between facts. The agent will learn some facts more rapidly than others, or may not be able to learn some facts by performing an associated task. If this parameter is set, then details should be provided in the task notes field on the class details page.
- *Learning-by-doing rate is different across time*: If this parameter is set, then the learning-by-doing rate is not constant for the duration of the simulation. In this case, the differences should be specified in the task notes field of the class details page.
- *Has communication network*: This parameter specifies whether an agent is part of a communication network, a bi-directional network for communication.
- *Has command network*: This parameter specifies whether an agent is part of a command network, a directed network for communication. Details about the command network should be provided in the tasks notes field of the class details page.

The message content table contains information about the types of information that the agent can communicate during an interaction.

- *Transmits facts*: This parameter specifies whether an agent can send or receive facts. If set, the agent is capable of transmitting facts but is not guaranteed to do so in any given interaction.
- *Transmits beliefs*: This parameter specifies whether an agent can send or receive beliefs. If set, the agent is capable of transmitting beliefs but is not guaranteed to do so in any given interaction.
- *Transmits who-knows-who transactive memory*: This parameter specifies whether an agent can send or receive who-knows-who transactive memory. If set, the agent is capable of transmitting this type of transactive memory but is not guaranteed to do so in any given interaction.
- *Transmits who-knows-what transactive memory*: This parameter specifies whether an agent can send or receive who-knows-what transactive memory. If set, the agent is capable of transmitting this type of transactive memory but is not guaranteed to do so in any given interaction.

- *Transmits who-does-what transactive memory*: This parameter specifies whether an agent can send or receive who-does-what transactive memory. If set, the agent is capable of transmitting this type of transactive memory but is not guaranteed to do so in any given interaction.
- *Transmits who-believes-what transactive memory*: This parameter specifies whether an agent can send or receive who-believes-what transactive memory. If set, the agent is capable of transmitting this type of transactive memory but is not guaranteed to do so in any given interaction.
- *Sends and Receives* This parameter specifies whether all of the transactive memory transmission parameters are bi-directional, meaning that the agent can both send and receive that form of transactive memory. Informational transmission need not be bi-directional, as agents are capable of being able to receive or create transactive memory information without being able to send it. The parameters do not all have the same value, this parameter should be set to ++ and further details should be provided in the interaction notes field on the class details page.
- *Transmits referrals*: This parameter specifies whether an agent can send or receive referrals. If set, the agent is capable of transmitting this type of transactive memory but is not guaranteed to do so in any given interaction.

The interaction characteristics table contains a number of parameters that directly govern the number of interactions that an agent can have during a given period. It also specifies how much information can be transmitted during these transactions.

- *Interactions per period*: This parameter, specified as a range with a min and a max, specifies the total number of interactions in which an agent of this class can participate during each simulation period. The total number of interactions each period is the sum of all initiated and received interactions, so limiting this number will either limit the number of initiations, receptions, or both. An agent that cannot interact with any agent should have a maximum interaction value of zero, while an agent that can interact with any number of agents should have a value of ++. Agents will never exceed more than their maximum number of interactions per period, but may not reach their minimum if their are insufficient agents with which to interact due to Construct's partner selection algorithm.
- *Initiations per period*: This parameter, specified as a range with a min and a max, specifies the total number of interactions that an agent of this class this agent will initiate during each simulation period. An agent that cannot initiate contact with any agent should have a maximum initiation value of zero. Agents will never exceed more than their maximum initiation count, but may not reach their minimum if there are insufficient agents that are able to receive communication.
- *Receptions per period*: This parameter, specified as a range with a min and a max, specifies the total number of interactions that an agent of this class can receive during each simulation period. Agents will never exceed more than their maximum reception count, but may not reach their minimum if there are insufficient agent who choose to initiate communication.
- *Unique interaction partners per period*: This parameter, specified as a range with a min and a max, specifies the total number of unique interaction partners per period. This quantity is different from the total number of interactions, as it may be possible for an agent to interact with the same agent more than once in a given time period. The minimum value of this parameter should be at least as great as the minimum number of interactions per period.
- *Message length*: This parameter, specified as a range with a min and a max, specifies the amount of information that an agent of this class can send during one interaction. This parameter only specifies the total message length, not the message composition; the types of information in the message is specified by parameters in the message content table.
- *Distinct messages sent*: This parameter, specified as a range with a min and a max, specifies the total number of messages that can be sent by an agent of this class. If the number of distinct messages sent is less than the number of interaction partners, some agents will be sent the same message.

The interaction patterns table specifies parameters that govern how an agent will go about its search for an interaction partner.

- *Interaction sphere density*: This parameter, specified as a range with a min and a max, indirectly specifies the number of agents with which agents of this class can interact as a function of the number of agents. The density of the sphere, times the number of agents in the simulation, will give the size of the interaction sphere. If specific agents, or classes of agents, are supposed to be included (or excluded) from the interaction sphere then the values should be set to ++ and details should be provided in the interaction notes table of the class details page.
- *Sphere is randomly generated*: If this parameter is set, then an agent's interaction sphere is generated randomly to be of the size specified. If unset, then the sphere should be fully specified in the interaction notes field of the class details page. If the value is set to ++, either the interaction sphere is formed one of the standard alternative methods or the details of the method should be specified in the interaction notes field of the class details page.
- *Sphere formed using socio-demographic similarity matching*: If this parameter is set, then the interaction sphere is created with a heavy emphasis on matching agents with those having similar socio-demographic settings.
- *Sphere formed using two-away network generator*: If this parameter is set, then the interaction sphere is formed using an ego net of size two. This means that the interaction sphere will be created using a pre-defined agent-agent matrix; any agent that is two links or fewer away from this agent will be included.
- *Interaction sphere size changes over time*: This parameter specifies whether the agents of this class have an interaction sphere size that remains constant throughout the virtual experiment. If unset, the sphere size is fixed though the actual composition of agents may change. If set, details about how the sphere expands or contracts should be provided in the interaction notes field on the class details page.
- *Interaction sphere composition changes over time*: This parameter specifies whether the agents in an agent's interaction sphere will remain constant throughout the experiment. If unset, the agent will have a constant set of interaction partners. If set, however, the agents in the sphere will change as some agents enter and others leave; the details as to which agents enter and leave should be provided in the interaction notes field on the class details page.
- *Agent can be isolated*: This parameter specifies whether agents of this class must be isolated or whether they are active during the entire simulation. If an agent is ever isolated, the details of this isolation should be provided in the active periods field on the class details page.
- *Percentage interactions by strategy*: These parameters govern how likely an agent of this class will be to engage in a particular interaction strategy. Agents will choose one of these strategies based upon the percentages specified, but will not be guaranteed to use a particular strategy during a given period. If these values do not sum to one hundred percent, a note should be provided in the interactions notes field on the class details page.
- *Fraction changes over time*: This parameter specifies whether the interaction strategy parameters will change over the course of a virtual experiment. If set, the details of this change should be provided in the interaction notes field on the class details page.

## B.2 Class details page

The second page of the agent class style sheet, the class details page, is divided into five text boxes. These fields provide additional information about some of the parameters specified in the class overview page. Some of these fields may be left blank if strictly unnecessary; however, some of these fields must be completed if certain overview fields have certain entries.

## B.3 Examples and Use

Class specification style sheets have been generated for Construct's default classes; these style sheets can be found in appendix C. Users who design their own classes are strongly encouraged to create these reports to document the important features of their classes. Future versions of Construct may eventually be able to automate the style sheet generation process, though as of this writing style-sheet generation is a human-intensive task. Time spent specifying an agent in a style sheet, however, is not time lost: by filling out a class style sheet, a simulation designer is able to better understand how class agents will interact in a virtual experiment.

# C Default Class Details

The following pages enumerate the many default agent classes commonly used in Construct simulations. These classes are designed for general-purpose applications, easy use, and rapid customization. Users may find these default agents especially helpful in the early stages of designing a virtual experiment, any may be useful jumping off points for the creation of customized classes.

For more information about the format of the class specification style sheets, specifically about the meaning of each of the fields and the type of information included in each field, see appendix B.

## C.1 Human (Simple)

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | No |
| Has initial beliefs | No |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | Yes |
| | |
| Can send beliefs | No |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 1 |
| Maximum | 1 |
| Minimum receptions / period | 1 |
| Maximum | 1 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 1 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | ++ |
| Maximum in total | ++ |
| | |
| Performs binary tasks | Yes |
| Performs energy tasks | Yes |
| | |
| Can be pre-assigned to tasks | Yes |
| Tasks change over time | Yes |
| Can choose task assignments | Yes |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 1.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | Yes |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Simple Human Agent Class (Continued)**

**Default Knowledge**

A simple human agent will have a 25% chance of knowing each of the simulated, non-task-specific facts. This does not guarantee that a particular agent will know 25% of these facts, as some agents may know more and some know less. The distribution of task-specific facts is left to the experiment designer. Agents of this class can learn any non-transactive memory, non-belief fact during the course of the simulation.

**Default Beliefs**

A simple human cannot have beliefs, and cannot learn belief facts during the simulation.

**Active Periods**

A simple human is generally active for all periods.

**Interaction Notes**

None for this agent class.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

Agents of this class can typically perform all possible tasks, unless otherwise specified by the simulation designer.

**Notes**

The simple human represents human without transactive memory or beliefs. This type of agent should be used if the simulation designer feels that beliefs and transactive memory are not necessary for the simulation, as these agents are smaller and may allow the simulation designer to double the number of agents simulated. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children.

## C.2 Human (Standard)

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | Yes |
| Has who-is-doing-what TM | No |
| Has beliefs TM | Yes |
| Has initial values | No |
| | |
| Follows referral suggestions | Yes |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | Yes |
| | |
| Can send beliefs | Yes |
| Can receive | Yes |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | Yes |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | Yes |
| Sends and receives | Yes |
| | |
| Can send referral | Yes |
| Can receive | Yes |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 1 |
| Maximum | 1 |
| Minimum receptions / period | 1 |
| Maximum | 1 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 1 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | ++ |
| Maximum in total | ++ |
| | |
| Performs binary tasks | Yes |
| Performs energy tasks | Yes |
| | |
| Can be pre-assigned to tasks | Yes |
| Tasks change over time | Yes |
| Can choose task assignments | Yes |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 1.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | Yes |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Human Agent Class (Continued)**

**Default Knowledge**

A standard human agent will have a 25% chance of knowing each of the simulated, non-task-specific facts. This does not guarantee that a particular agent will know 25% of these facts, as some agents may know more and some know less. The distribution of task-specific facts is left to the experiment designer. Agents of this class can learn any fact during the course of the simulation, including beliefs and transactive memory facts.

**Default Beliefs**

Agents of this class will have a 10% chance of knowing each of the negative belief facts in order to introduce some variance into the population. This parameter shuold be changed depending upon the type of simulation being performed.

**Active Periods**

A standard human agent is generally active for all periods.

**Interaction Notes**

None for this agent class.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

Agents of this class can typically perform all possible tasks, unless otherwise specified by the simulation designer.

**Notes**

The standard human represents human with fact knowledge, transactive memory, and beliefs. It is the workhorse agent class of most Construct simulations, as most virtual experiments are designed to see how an intervention will affect the decisions made by agents of this class. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children.

## C.3 Human with Transactive Memory

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | No |
| Has initial beliefs | No |
| | |
| Has who-knows-who TM | Yes |
| Has who-knows-what TM | Yes |
| Has who-is-doing-what TM | Yes |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | Yes |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | Yes |
| | |
| Can send beliefs | No |
| Can receive | No |
| | |
| Transmits who-knows-who TM | Yes |
| Transmits who-knows-what TM | Yes |
| Transmits who-does-what TM | Yes |
| Transmits beliefs TM | No |
| Sends and receives | Yes |
| | |
| Can send referral | Yes |
| Can receive | Yes |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 1 |
| Maximum | 1 |
| Minimum receptions / period | 1 |
| Maximum | 1 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 1 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | ++ |
| Maximum in total | ++ |
| | |
| Performs binary tasks | Yes |
| Performs energy tasks | Yes |
| | |
| Can be pre-assigned to tasks | Yes |
| Tasks change over time | Yes |
| Can choose task assignments | Yes |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 1.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | Yes |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Human with Transactive Memory Agent Class (Continued)**

| **Default Knowledge** |
| --- |
| A standard human agent will have a 25% chance of knowing each of the simulated, non-task-specific facts. This does not guarantee that a particular agent will know 25% of these facts, as some agents may know more and some know less. The distribution of task-specific facts is left to the experiment designer. Agents of this class can learn any non-belief memory fact during the course of the simulation. |

| **Default Beliefs** |
| --- |
| Agents of this class cannot hold beliefs, and cannot learn belief facts during the simulation. |

| **Active Periods** |
| --- |
| This agent class generally is active for all periods. |

| **Interaction Notes** |
| --- |
| None for this agent class. |

| **Internal Characteristic Notes** |
| --- |
| None for this agent class. |

| **Task Notes** |
| --- |
| Agents of this class can typically perform all possible tasks, unless otherwise specified by the simulation designer. |

| **Notes** |
| --- |
| The human with transactive memory agent class represents a human with knowledge, who-knows-who transactive memory, who-knows-what transactive memory, and who-does-what transactive memory; however, agents of this class differ from standard humans because they cannot develop beliefs. Agents of this class should be used when beliefs are irrelevant to the simulation for simulation efficiency. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children. |

## C.4 Human with Beliefs

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | Yes |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | Yes |
| | |
| Can send beliefs | Yes |
| Can receive | Yes |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | Yes |
| Can receive | Yes |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 1 |
| Maximum | 1 |
| Minimum receptions / period | 1 |
| Maximum | 1 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 1 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | ++ |
| Maximum in total | ++ |
| | |
| Performs binary tasks | Yes |
| Performs energy tasks | Yes |
| | |
| Can be pre-assigned to tasks | Yes |
| Tasks change over time | Yes |
| Can choose task assignments | Yes |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 1.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | Yes |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Human with Beliefs Agent Class (Continued)**

| **Default Knowledge** |
| --- |
| A human agent with beliefs will have a 25% chance of knowing each of the simulated, non-task-specific facts. This does not guarantee that a particular agent will know 25% of these facts, as some agents may know more and some know less. The distribution of task-specific facts is left to the experiment designer. Agents of this class can learn any non-transactive memory fact during the course of the simulation. |

| **Default Beliefs** |
| --- |
| Agents of this class will have a 10% chance of knowing each of the negative belief facts in order to introduce some variance into the population. This parameter shuold be changed depending upon the type of simulation being performed. |

| **Active Periods** |
| --- |
| Agents of this class are generally active for all periods by default. |

| **Interaction Notes** |
| --- |
| None for this agent class. |

| **Internal Characteristic Notes** |
| --- |
| None for this agent class. |

| **Task Notes** |
| --- |
| Agents of this class can typically perform all possible tasks, unless otherwise specified by the simulation designer. |

| **Notes** |
| --- |
| The human with beliefs agent class represents a human with knowledge and beliefs, and will be able to act on their beliefs when making a decision. Unlike agents of the standard human class, however, they lack transactive memory; this makes them more memory efficient in simulations that choose not to model transactive memory. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children. |

## C.5 Human with Mass Media

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | Yes |
| Has who-is-doing-what TM | No |
| Has beliefs TM | Yes |
| Has initial values | No |
| | |
| Follows referral suggestions | Yes |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | Yes |
| | |
| Can send beliefs | Yes |
| Can receive | Yes |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | Yes |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | Yes |
| Sends and receives | Yes |
| | |
| Can send referral | Yes |
| Can receive | Yes |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 1 |
| Maximum | ++ |
| Minimum receptions / period | 1 |
| Maximum | 1 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 1 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | ++ |
| Maximum in total | ++ |
| | |
| Performs binary tasks | Yes |
| Performs energy tasks | Yes |
| | |
| Can be pre-assigned to tasks | Yes |
| Tasks change over time | Yes |
| Can choose task assignments | Yes |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 1.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | Yes |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Human with Mass Media Agent Class (Continued)**

**Default Knowledge**

A human agent with access to mass media will have a 15% chance of knowing each of the simulated, non-task-specific facts. This does not guarantee that a particular agent will know 15% of these facts, as some agents may know more and some know less. The distribution of task-specific facts is left to the experiment designer. Agents of this class can learn any fact during the course of the simulation. Note that the amount of standard knowledge is slightly less than the amount known by standard humans, meaning that standard agents acting purely on the basis of homophily will be less likely to choose to interact with this agent.

**Default Beliefs**

Agents of this class will have a 10% chance of knowing each of the negative belief facts in order to introduce some variance into the population. Because humans with mass media are usually used as interventions, however, a simulation designer may find it worthwhile to explicitly set the beliefs of this agent.

**Active Periods**

Agents of this class typically are used in interventions, and thus are active only intermittently. Frequently, such agents will be active only one or two interaction periods in an entire simulation.

**Interaction Notes**

Agents of this class can initiate communication with potentially every other agent in the simulation.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

None for this agent class.

**Notes**

Agents of the human with mass media class are standard human agents who are able to communicate with many agents simultaneously, usually using a transmission mechanism such as radio or television. This mass media allows them to send their information and beliefs much more widely than other human agents, but this property may make it difficult or impossible for these agents to receive new information. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, humans with mass media may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents.

## C.6 Book

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | No |
| Has initial beliefs | No |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | No |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 0.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 0.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 0.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 0 |
| Maximum | ++ |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 5 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Book Agent Class (Continued)**

**Default Knowledge**

A book's initial knowledge is usually fixed at five facts; these facts remain fixed during the simulation. Books do not learn or forget facts over time, and lack any form of transactive memory.

**Default Beliefs**

Books are meant to be repositories of knowledge and do not have beliefs. However, it is possible for books to be assigned belief facts, and agents who choose to read the book may change their beliefs as a consequence of learning these facts.

**Active Periods**

Agents of this class can either be used as agents or as interventions. Generally, these agents are active every time period but few other agents will choose to interact with them.

**Interaction Notes**

Agents of the book class cannot initiate communication with other agents, but can receive communication from multiple other agents during a round. Generally, the number of agents with which a book can interact is bounded by the simulation designer, though many simulations will assume that there are an unlimited copies of the book and any agent will be able to read it instead of interacting with another human agent. Books, because of their focus on an issue, typically convey more information than humans during a particular interaction.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

Agents of this class cannot perform tasks.

**Notes**

Agents of the book class represent repositories of written knowledge; when agents spend an interaction period with an agent of the book class they can be said to be reading. Construct assumes that there are many copies of a book by default, so it is possible for multiple agents to interact with the same book agent during one period. Books do not have express beliefs, though they can contain belief facts. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, books may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents.

## C.7 Leaflet

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | No |
| Has initial beliefs | No |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | No |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 0.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 0.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 0.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 0 |
| Maximum | ++ |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 1 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Leaflet Agent Class (Continued)**

| **Default Knowledge** |
| --- |
| A leaflet's initial knowledge is fixed at one fact; this fact remains fixed during the simulation. Leaflets cannot learn or forget facts, and do not have any form of beliefs or transactive memory. |

| **Default Beliefs** |
| --- |
| Leaflets convey information only and do not have beliefs. |

| **Active Periods** |
| --- |
| Leaflets are generally only active during specific time periods of a simulation, and these periods should be specified by the simulation designer. |

| **Interaction Notes** |
| --- |
| Agents of the leaflet class cannot initiate communication with other agents, but can receive communication from multiple other agents during a round. Leaflets typically convey less information than humans during a particular interaction. |

| **Internal Characteristic Notes** |
| --- |
| None for this agent class. |

| **Task Notes** |
| --- |
| Agents of this class cannot perform tasks. |

| **Notes** |
| --- |
| Agents of the leaflet class represent posters designed to quickly convey a belief or knowledge fact; leaflets contain much less information than books do. Construct assumes that there are many copies of a leaflet by default, so it is possible for multiple agents to interact with the same leaflet agent during the same period. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, leaflets may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents. |

## C.8 Flier

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | Yes |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 0.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 0.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 0.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 0 |
| Maximum | ++ |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 1 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Flier Agent Class (Continued)**

| **Default Knowledge** |
| --- |
| A flier's initial knowledge is usually fixed at one fact; this fact remains fixed during the simulation. Fliers cannot gain or forget information between interaction periods, and do not have any form of transactive memory. |

| **Default Beliefs** |
| --- |
| Fliers, unlike leaflets, are meant to convey a fact or an opinion very quickly. Fliers can have one belief, and this belief cannot be changed during the course of the simulation. |

| **Active Periods** |
| --- |
| Fliers are generally only active during specific time periods of a simulation, and these periods should be specified by the simulation designer. |

| **Interaction Notes** |
| --- |
| Agents of the flier class cannot initiate communication with other agents, but can receive communication from multiple other agents during a round. Generally, the number of agents with which a flier can interact is unbounded. Fliers typically convey less information than humans during a particular interaction. |

| **Internal Characteristic Notes** |
| --- |
| None for this agent class. |

| **Task Notes** |
| --- |
| Agents of this class cannot perform tasks. |

| **Notes** |
| --- |
| Agents of the flier class represent posters designed to quickly convey a belief or knowledge fact; unlike leaflets, fliers have beliefs as well as facts. Construct assumes that there are many copies of a flier by default, so it is possible for multiple agents to interact with the same flier agent during the same period. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, fliers may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents. |

## C.9 Web site

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | Yes |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | Yes |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 0.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 0.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 0.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 0 |
| Maximum | ++ |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 5 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Web Site Agent Class (Continued)**

**Default Knowledge**

A web site's initial knowledge is fixed at five facts. Web sites cannot learn or forget information, and all current implementations of web sites lack transactive memory. Future extensions to the website class will include websites that can refer agents to other agents who can perform specific tasks or knowledge; however, these agents have not been implemented as of this writing.

**Default Beliefs**

Web sites may have beliefs, depending upon the needs of the simulation designer; if so, these beliefs should be specified by the simulation designers. These beliefs may be stored in terms of an initial belief or may be codified in terms of belief facts which can be transmitted to other agents. If a website has a belief, that belief will remain constant throughout the simulation since the website is assumed to be static.

**Active Periods**

Web sites are generally active for all simulation periods, though in some simulations it may be advantageous to have web sites appear cyclically or appear after a specific period.

**Interaction Notes**

Agents of the web site class cannot initiate communication with other agents, but can receive communication from multiple other agents during a round. Web sites, because of their particular focus on an issue, typically convey more information than humans during a particular interaction.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

Agents of this class cannot perform tasks.

**Notes**

Agents of the web site class represent online web sites of particular relevance to the simulation; when agents spend an interaction period with an agent of the web site class they can be thought of as browsing for online information relevant to a particular subject. Construct assumes that web site use is non-exclusive, so it is possible for multiple agents to interact with the same web site agent during one period. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, fliers may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents.

## C.10 Task Database

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | ++ |
| | |
| Has beliefs | No |
| Has initial beliefs | No |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | Yes |
| | |
| Can send beliefs | No |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 0.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 0.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 0.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 0 |
| Maximum | ++ |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 5 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | No |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Task Database Class (Continued)**

| **Default Knowledge** |
| --- |
| A task database usually lacks default knowledge and begins the virtual experiment empty; in certain experiments, however, the designer may choose to prime the database with a few facts. Over time, as agents interact with the database, they may expand it by adding additional information. Task databases can only store fact information, and do not store any transactive memory elements. |

| **Default Beliefs** |
| --- |
| Agents of the task database class do not have beliefs, though they can pass belief facts to the agents with whom they interact. |

| **Active Periods** |
| --- |
| Task databases are generally active for all simulation periods, though in some simulations task databases may be isolated. |

| **Interaction Notes** |
| --- |
| Agents of the task database class cannot initiate communication with other agents, but can receive communication from multiple other agents during a round. Task databases, because of their focus on the simulated issue, typically convey more information on that issue than humans do in a particular interaction round, as it is assumed that individuals interacting with the task database do so in order to find out more information on the particular issue. |

| **Internal Characteristic Notes** |
| --- |
| None for this agent class. |

| **Task Notes** |
| --- |
| Agents of this class cannot perform tasks. |

| **Notes** |
| --- |
| Agents of the task database class represent repositories of knowledge about tasks and facts; when agents spend an interaction period with a task database agent they can be said to be reading or writing information to the database. Construct assumes that multiple agents can interact with the same referential database simultaneously during one period. Since the additions to the database are not made available until the end of the round, the order of interaction does not matter. Agents of the task database class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children. Due to overlaps in the socio-demographics of the target audience, task databases may have multiple values for the same socio-demographic attribute. |

## C.11 Referential Database

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | ++ |
| | |
| Has beliefs | No |
| Has initial beliefs | No |
| | |
| Has who-knows-who TM | Yes |
| Has who-knows-what TM | Yes |
| Has who-is-doing-what TM | Yes |
| Has beliefs TM | Yes |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | Yes |
| | |
| Can send beliefs | No |
| Can receive | No |
| | |
| Transmits who-knows-who TM | Yes |
| Transmits who-knows-what TM | Yes |
| Transmits who-does-what TM | Yes |
| Transmits beliefs TM | Yes |
| Sends and receives | No |
| | |
| Can send referral | Yes |
| Can receive | Yes |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 0.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 0.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 0.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 0 |
| Maximum | ++ |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 5 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | No |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Referential Database Class (Continued)**

**Default Knowledge**

A referential database usually lacks default knowledge and begins the virtual experiment empty; in certain experiments, however, the designer may choose to prime the database with a few facts or references. Over time, as agents interact with the database, they may expand it by adding additional information. Unlike task databases, referential databases can store transactive facts as well: it can store who-knows-who, who-knows-what, who-does-what, and who-believes-what transactive memory.

**Default Beliefs**

Agents of the referential database class do not have beliefs, though they can pass belief facts to the agents with whom they interact. The referential database can also store beliefs transactive memory about other agents.

**Active Periods**

Referential databases are generally active for all simulation periods, though in some simulations referential databases may be isolated.

**Interaction Notes**

Agents of the referential database class cannot initiate communication with other agents, but can receive communication from multiple other agents during a round. Referential databases, because of their focus on the simulated issue, typically convey more information on that issue than humans do in a particular interaction round, as it is assumed that individuals interacting with the task database do so in order to find out more information on the particular issue.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

Agents of this class cannot perform tasks.

**Notes**

Agents of the referential database class represent repositories of knowledge about other agents; when agents spend an interaction period with a referential database agent they can be said to be reading or writing information to the database. A referential database is a task database that can store and retrieve transactive memory as well. Construct assumes that multiple agents can interact with the same referential database simultaneously during one period. Since the additions to the database are not made available until the end of the round, the order of interaction does not matter. Agents of the referential database class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, referential databases may have multiple values for the same socio-demographic attribute.

## C.12 Avatar

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | ++ |
| Has initial knowledge | ++ |
| | |
| Has beliefs | No |
| Has initial beliefs | No |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | No |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 0.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 0.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 0.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 0.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 0 |
| Maximum | ++ |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 1 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | No |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Avatar Class (Continued)**

| **Default Knowledge** |
| --- |
| Since an avatar is linked to a human agent, an avatar's knowledge is linked to the knowledge of the associated human as well. As the human gains or loses knowledge, the avatar gains or loses knowledge as well; avatars cannot gain or lose facts independently of their associated agent. Avatars do not have any form of transactive memory. |

| **Default Beliefs** |
| --- |
| Agents of the avatar class do not have beliefs, though they can pass belief facts to the agents with whom they interact. |

| **Active Periods** |
| --- |
| Avatars are generally active for all simulation periods, though in some simulations they may be isolated. |

| **Interaction Notes** |
| --- |
| Agents of the avatar class cannot initiate communication with other agents, but they can receive communication if contacted by multiple other agents during a round. Avatars can only convey one fact during an interaction. |

| **Internal Characteristic Notes** |
| --- |
| None for this agent class. |

| **Task Notes** |
| --- |
| Agents of this class cannot perform tasks, though their associated human agents will be able to do so. |

| **Notes** |
| --- |
| Agents of the avatar class represent repositories of knowledge for an associated human agent. Agents who wish to learn knowledge from the human agent may be able to speak to the avatar to learn this information, and since the avatar can communicate with multiple agents during a round it may help facilitate information flow. Avatars can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, avatars may have multiple values for the same socio-demographic attribute. |

## C.13 Promoter

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | Yes |
| Can receive | ++ |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 1 |
| Maximum | 1 |
| Minimum receptions / period | 1 |
| Maximum | 4 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 3 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | ++ |
| Maximum in total | ++ |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 1.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | No |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | Yes |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Promoter Agent Class (Continued)**

**Default Knowledge**

An agent of the promoter class will have a 100% chance of knowing each fact related to a particular tax scheme, though it may have access to other facts as well. Additionally, a promoter has full knowledge of 25% of all other simulated facts, and can learn new knowledge facts during the course of the simulation.

**Default Beliefs**

A promoter agent is designed to sway beliefs and encourage scheme participation, and as such will have a 100% chance of knowing each pro-scheme legality fact and pro-scheme participation fact.

**Active Periods**

Though agents of this class typically are used in interventions, they tend to be more active than other interventions such as advertisements or seminars. Frequently, such agents will be active every other interaction periods in a simulation, which is less frequently than the standard agents but more frequently than some of the more powerful interventions.

**Interaction Notes**

The promoter's interaction sphere is usually specified by the simulation designer. Different virtual experiments have focused on the roles of different promoter interaction spheres, such as the effect of sphere overlap when multiple promoters are present.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

This agent cannot perform tasks.

**Notes**

Promoter agents are designed to propagate scheme-specific information to other agents in the simulation. The promoter agent class is similar to a human agent with beliefs class, though they cannot learn new belief facts. However, they differ from agents of that class because they can communicate with multiple agents per period and can transmit longer messages. Agents of this the promoter class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, promoters may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents.

## C.14 Seminar

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | No |
| Has initial beliefs | No |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | No |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 5 |
| Maximum | 5 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 13 |
| Maximum | 13 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | No |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | Yes |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Seminar Agent Class (Continued)**

| **Default Knowledge** |
| --- |
| An agent of the seminar class will have a 100% chance of knowing each fact related to a particular tax scheme, though it may have access to other facts as well. Additionally, a seminar has full knowledge of 10% of all other simulated facts. In contrast to promoters, seminar agents cannot learn new facts during the simulation. |

| **Default Beliefs** |
| --- |
| A seminar agent is designed to sway beliefs and encourage scheme participation, and as such will have a 100% chance of knowing each pro-scheme legality fact and pro-scheme participation fact. |

| **Active Periods** |
| --- |
| Agents of this class typically are used in interventions, and thus are active only intermittently. Frequently, such agents will be active only one or two interaction periods in an entire simulation; however, seminars will usually occur more frequently than short courses and will often be held several simulated months before the date on which taxes are due. |

| **Interaction Notes** |
| --- |
| A seminar cannot initiate interactions, but can receive interactions from many agents. The seminar's interaction sphere is usually specified by the simulation designer. |

| **Internal Characteristic Notes** |
| --- |
| None for this agent class. |

| **Task Notes** |
| --- |
| This agent cannot perform tasks. |

| **Notes** |
| --- |
| The seminar agent is designed to spread large amounts of information about a particular scheme, and to spread this information to multiple people at a time. Seminars, like promoters, can only receive communication, but they can communicate with more agents every period also can transmit longer messages when they do so. Seminars, though, are active less frequently than promoters; however, when they are active, they can transmit much more information. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, seminars may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents. |

## C.15 Short Course

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | Yes |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 5 |
| Maximum | 5 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | ++ |
| Maximum | ++ |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | No |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | Yes |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Short Course Agent Class (Continued)**

| **Default Knowledge** |
| --- |
| An agent of the short course class will have a 100% chance of knowing each fact related to a particular tax scheme, though it may have access to other facts as well. Additionally, a seminar has full knowledge of 10% of all other simulated facts. In contrast to promoters, short course agents cannot learn new facts during the simulation. |

| **Default Beliefs** |
| --- |
| A short course agent is designed to sway beliefs and encourage scheme participation, and as such will have a 100% chance of knowing each pro-scheme legality fact and pro-scheme participation fact. |

| **Active Periods** |
| --- |
| Agents of this class typically are used in interventions, and thus are active only intermittently. Frequently, such agents will be active only one or two interaction periods in an entire simulation. |

| **Interaction Notes** |
| --- |
| A short course cannot initiate interactions, but can receive interactions from many agents. The short course's interaction sphere is usually specified by the simulation designer. |

| **Internal Characteristic Notes** |
| --- |
| None for this agent class. |

| **Task Notes** |
| --- |
| This agent cannot perform tasks. |

| **Notes** |
| --- |
| The short course agent is designed to spread large amounts of information about a particular scheme. Short courses, like seminars, can only receive communication, but they can communicate with more agents every period also can transmit longer messages when they do so. Short courses, though, are active even less frequently than seminars; however, when they are active, they can transmit much more information than seminars. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, seminars may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents. |

## C.16 Pro-Scheme Advertisement

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | Yes |
| Can receive | Yes |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | No |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 1 |
| Maximum | 20 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 5 |
| Maximum | 5 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | Yes |
| S-D similarity matching | No |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | Yes |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Pro-Scheme Advertisement Agent Class (Continued)**

**Default Knowledge**

An agent of the pro-scheme advertisement class will have a 10% chance of knowing each fact related to a particular tax scheme, though it may have access to other facts as well. This does not mean that every advertisement will have 10% of the total facts, as some advertisements may have more and some will have less than others. During the simulation, a pro-scheme advertisement cannot learn new facts, receive new facts, or create transactive memory.

**Default Beliefs**

A pro-scheme advertisement is designed to sway beliefs and encourage scheme participation, and as such will have a 75% chance of knowing each pro-scheme legality fact and pro-scheme participation fact. Additionally, the prior belief of the advertisement can be explicitly set to increase the strength of the influentialness of the advertisement.

**Active Periods**

Agents of this class typically are used in interventions, and thus are active only intermittently. Frequently, such agents will be active only one or two interaction periods in an entire simulation.

**Interaction Notes**

An advertisement cannot initiate interactions, but can receive interactions from many agents. The advertisement's interaction sphere is usually specified by the simulation designer, and designers can choose whether the ad is targeted to a specific audience or available for the general public. An advertisement can interact with up to twenty other agents in a simulation round and can exchange up to five facts.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

This agent cannot perform tasks.

**Notes**

The pro-scheme advertisement agent is designed to spread small amounts of information about a particular scheme. Advertisements can only receive communication, but they can communicate with many more agents than a promoter or seminar during a given interaction round. Pro-scheme advertisements are designed to sway beliefs and have only minimal knowledge impact. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, pro-scheme advertisements may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents.

## C.17 Anti-Scheme Advertisement

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | Yes |
| Can receive | Yes |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | No |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 1 |
| Maximum | 20 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 5 |
| Maximum | 5 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | Yes |
| S-D similarity matching | No |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | Yes |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Anti-Scheme Advertisement Agent Class (Continued)**

**Default Knowledge**

An agent of the anti-scheme advertisement class will have a 10% chance of knowing each fact related to a particular tax scheme, though it may have access to other facts as well. This does not mean that every advertisement will have 10% of the total facts, as some advertisements may have more and some will have less than others. Additionally, an agent of the anti-scheme advertisement class will have a 30% chance of knowing each fact related to legally filing one's taxes, though again this does not mean that each advertisement will have 30% of the total facts. During the simulation, an advertisement cannot learn new facts, receive new facts, or create transactive memory.

**Default Beliefs**

A anti-scheme advertisement is designed to sway beliefs and encourage scheme participation, and as such will have a 90% chance of knowing each anti-scheme legality fact and anti-scheme participation fact. Additionally, the prior belief of the advertisement can be explicitly set to increase the strength of the influentialness of the advertisement.

**Active Periods**

Agents of this class typically are used in interventions, and thus are active only intermittently. Frequently, such agents will be active only one or two interaction periods in an entire simulation.

**Interaction Notes**

An advertisement cannot initiate interactions, but can receive interactions from many agents. The advertisement's interaction sphere is usually specified by the simulation designer, and designers can choose whether the ad is targeted to a specific audience or available for the general public. An advertisement can interact with up to twenty other agents in a simulation round and can exchange up to five facts.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

This agent cannot perform tasks.

**Notes**

The anti-scheme advertisement agent is designed to discourage participation in tax schemes in general. Advertisements can only receive communication, but they can communicate with many more agents than a promoter or seminar during a given interaction round. Anti-scheme advertisements are designed to sway beliefs and have only minimal knowledge impact. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, anti-scheme advertisements may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents.

## C.18 Tax Assistance Center

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | No |
| Has who-is-doing-what TM | No |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | No |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | Yes |
| Can receive | No |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | No |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | No |
| Can receive | No |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 0 |
| Maximum | 0 |
| Minimum receptions / period | 5 |
| Maximum | 5 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 13 |
| Maximum | 13 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | 0 |
| Maximum in total | 0 |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | No |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | ++ |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Assistance Center Agent Class (Continued)**

**Default Knowledge**

An agent of the anti-scheme advertisement class will have a 10% chance of knowing each fact related to a particular tax scheme, though it may have access to other facts as well. This does not mean that every advertisement will have 10% of the total facts, as some advertisements may have more and some will have less than others. Additionally, an agent of the anti-scheme advertisement class will have a 90% chance of knowing each fact related to legally filing one's taxes, though again this does not mean that each advertisement will have 90% of the total facts. During the simulation, an advertisement cannot learn new facts, receive new facts, or create transactive memory.

**Default Beliefs**

Though the primary goal of an assistance center is to file the taxes of an individual agent, it is also designed to sway beliefs and discourage scheme participation. For this reason, in addition to its other functions, an assistance center agent will have a 90% chance of knowing each anti-scheme legality fact and anti-scheme participation fact.

**Active Periods**

Agents of this class typically are used in interventions, and thus are active only intermittently. Frequently, such agents will be active only one or two interaction periods in an entire simulation.

**Interaction Notes**

An assistance center cannot initiate interactions, but can receive interactions from many agents. The center's interaction sphere is usually specified by the simulation designer.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

This agent cannot perform tasks.

**Notes**

The tax assistance center has a dual role. First, agents who interact with a tax assistance center are assumed to have correctly filed their taxes and will not be eligible to participate in a scheme. Additionally, though, the tax assistance center agent is designed to spread large amounts of information about how to legally complete tax forms and to come into compliance with tax laws. Tax centers, like auditors, can only receive communication, but they can communicate with more agents every period also can transmit more powerful messages when they do so. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, assistance centers may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents.

## C.19 Auditor

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | Yes |
| Has who-knows-what TM | Yes |
| Has who-is-doing-what TM | Yes |
| Has beliefs TM | No |
| Has initial values | No |
| | |
| Follows referral suggestions | Yes |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | No |
| | |
| Can send beliefs | Yes |
| Can receive | No |
| | |
| Transmits who-knows-who TM | Yes |
| Transmits who-knows-what TM | Yes |
| Transmits who-does-what TM | Yes |
| Transmits beliefs TM | No |
| Sends and receives | No |
| | |
| Can send referral | Yes |
| Can receive | Yes |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 1 |
| Maximum | 1 |
| Minimum receptions / period | 1 |
| Maximum | 1 |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 3 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | ++ |
| Maximum in total | ++ |
| | |
| Performs binary tasks | No |
| Performs energy tasks | No |
| | |
| Can be pre-assigned to tasks | No |
| Tasks change over time | No |
| Can choose task assignments | No |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 0.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | Yes |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | ++ |
| Isolation occurs randomly | ++ |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Auditor Agent Class (Continued)**

**Default Knowledge**

An auditor agent is designed to bring agents into compliance with tax law. As such, he has a 90% chance of knowing each of the compliance facts and 20% of the scheme facts; auditors also have access to 25% of the general facts as well. Since auditors are meant to represent interventions rather than human agents, they cannot learn new facts, create new transactive memories, or change their beliefs during the simulation.

**Default Beliefs**

Though the primary goal of an auditor is to audit the taxes of an individual agent, it is also designed to sway beliefs and discourage scheme participation in the future. For this reason, an auditor agent will have a 90% chance of knowing each anti-scheme legality fact and anti-scheme participation fact.

**Active Periods**

Agents of this class typically are used in interventions, and thus are active only intermittently. Frequently, such agents will be active only one or two interaction periods in an entire simulation.

**Interaction Notes**

None for this agent class.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

Agents of this class cannot perform tasks.

**Notes**

The auditor represents an internal revenue services auditor who encourages compliance with tax laws. Auditors can have initial transactive memory and beliefs, but these cannot be changed during the simulation. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, auditors may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents.

## C.20 Opinion Leader

| Knowledge & Beliefs | |
|---|---|
| Has fact knowledge | Yes |
| Has initial knowledge | Yes |
| | |
| Has beliefs | Yes |
| Has initial beliefs | Yes |
| | |
| Has who-knows-who TM | No |
| Has who-knows-what TM | Yes |
| Has who-is-doing-what TM | No |
| Has beliefs TM | Yes |
| Has initial values | No |
| | |
| Follows referral suggestions | Yes |
| Has initial values | No |

| Message Content | |
|---|---|
| Can send facts | Yes |
| Can receive | Yes |
| | |
| Can send beliefs | Yes |
| Can receive | Yes |
| | |
| Transmits who-knows-who TM | No |
| Transmits who-knows-what TM | Yes |
| Transmits who-does-what TM | No |
| Transmits beliefs TM | Yes |
| Sends and receives | ++ |
| | |
| Can send referral | Yes |
| Can receive | Yes |

| Internal Characteristics | |
|---|---|
| Minimum influentialness rate | 0.0 |
| Maximum | 1.0 |
| Minimum influenceability rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum attention rate | 0.0 |
| Maximum | 1.0 |
| Minimum forgetting rate | 0.0 |
| Maximum | 1.0 |
| | |
| Minimum risk aversion rate | 0.0 |
| Maximum | 1.0 |
| Has socio-demographics | Yes |
| Has location | Yes |

| Interaction Characteristics | |
|---|---|
| Minimum interactions / period | ++ |
| Maximum | ++ |
| Minimum initiations / period | 1 |
| Maximum | 1 |
| Minimum receptions / period | 0 |
| Maximum | ++ |
| | |
| Minimum unique / period | 1 |
| Maximum | ++ |
| | |
| Minimum length of message | 1 |
| Maximum | 7 |
| Minimum distinct messages sent | 1 |
| Maximum | 1 |

| Tasks & Performance | |
|---|---|
| Minimum number of tasks | 0 |
| Maximum simultaneously | ++ |
| Maximum in total | ++ |
| | |
| Performs binary tasks | Yes |
| Performs energy tasks | Yes |
| | |
| Can be pre-assigned to tasks | Yes |
| Tasks change over time | Yes |
| Can choose task assignments | Yes |
| | |
| Minimum task learning rate | 0.0 |
| Maximum | 1.0 |
| Different rate across tasks | No |
| Different rate across time | No |
| | |
| Has communication network | Yes |
| Has command network | No |

| Interaction Patterns | |
|---|---|
| Minimum sphere density | 0.0 |
| Maximum | 1.0 |
| | |
| Randomly generated sphere | ++ |
| S-D similarity matching | Yes |
| 2-away network generator | No |
| | |
| Sphere size changes over time | No |
| Membership changes over time | No |
| | |
| Agent can be isolated | Yes |
| Isolation is cyclical | No |
| Isolation occurs randomly | No |
| | |
| % interactions using homophily | 100 |
| % interactions deliberate search | 0 |
| % interactions with co-workers | 0 |
| Fraction changes over time | No |

**Opinion Leader Agent Class (Continued)**

**Default Knowledge**

An opinion leader will have a 15% chance of knowing each of the simulated, non-task-specific facts. This does not guarantee that a particular agent will know 15% of these facts, as some agents may know more and some know less. The distribution of task-specific facts is left to the experiment designer, as these are the major intervention effects of the opinion leader agent. Agents of this class can send knowledge and transactive memory but cannot receive it. Note that the amount of standard knowledge is slightly less than the amount known by standard humans, meaning that standard agents acting purely on the basis of homophily will be less likely to choose to interact with this agent.

**Default Beliefs**

Opinion leaders have beliefs that are usually fixed. Typically, opinion leaders will have strong beliefs and will have the belief facts that support their beliefs, though other opinion leaders may not have this characteristic. Opinion leaders also have beliefs transactive memory, and can send transmit beliefs transactive memory in their messages.

**Active Periods**

Agents of this class typically are used in interventions, and thus are active only intermittently. Frequently, such agents will be active only one or two interaction periods in an entire simulation.

**Interaction Notes**

Opinion leaders will be able to interact with many human agents due to their access to mass media.

**Internal Characteristic Notes**

None for this agent class.

**Task Notes**

None for this agent class.

**Notes**

Agents of the opinion leader class represent a person of note within a community and one whose beliefs carry some weight. These agents typically have access to mass media in order to publicize their beliefs and to spread their knowledge to multiple agents. Agents of this class can have the following socio-demographic characteristics: gender, age, income level, education level, race, and number of children; due to overlaps in the socio-demographics of the target audience, however, opinion leaders may be given multiple values for the same socio-demographic attribute in order to increase homophily with different agents.

# References


[1] Kathleen Carley. On the evolution of social and organizational networks. pages 1–29, 1999.

[2] Craig Schreiber, Siddhartha Singh, and Kathleen Carley. Construct - a multi-agent network model for the co-evolution of agents and socio-cultural environments. Technical report, Carnegie Mellon University School of Computer Science, May 2004.

[3] Kathleen Carley. Begining to end – construct. Unpublished Slide Presentation, Carnegie Mellon University School of Computer Science, 2005.

[4] Kathleen Carley. Smart agents and organizations of the future. *The Handbook of New Media*, 2002.

[5] Kathleen Carley. Model information update – 10-26-05. Unpublished Slide Presentation, Carnegie Mellon University School of Computer Science, 2005.

[6] Innovative Decisions Inc. Opera phase II final report. Unpublished Report, Carnegie Mellon University School of Computer Science, 2005.

[7] Kathleen Carley. Group stability: A socio-cognitive approach. *Advances in Group Processes*, 7:1–44, 1990.

[8] Kathleen Carley. Communication technologies and their effect on cultural homogeneity, consensus, and the diffusion of new ideas. *Sociological Perspectives*, 38(4):547–571, 1995.

[9] Craig Schreiber and Kathleen Carley. The impact of databases on knowledge transfer: simulation providing theory. In *NAACSOS Conference Proceedings*, Pittsburgh, PA, 2003.

[10] Kathleen Carley. Model information update – 12-06-05. Unpublished Slide Presentation, Carnegie Mellon University School of Computer Science, 2005.

[11] Mike Kowalchuck and Kathleen Carley. Understanding construct. Technical report, Carnegie Mellon University School of Computer Science, 2006.

[12] Kathleen Carley and Jeff Reminga. Ora: Organizational risk analyzer. Technical report, Carnegie Mellon University School of Computer Science, 2004.

[13] Jana Diesner and Kathleen Carley. Automap 1.2 - extract, analyze, represent, and compare mental models from texts. Technical report, Carnegie Mellon University School of Computer Science, 2004.